
MOTECH Documentation

Release 0.24-SNAPSHOT

Grameen Foundation

June 14, 2015

1	About MOTECH	3
1.1	MOTECH Architecture Overview	3
1.2	What can MOTECH do?	3
2	Creating Applications	5
2.1	Introduction	5
2.2	Installing and Upgrading MOTECH	7
2.3	Modeling Data with MOTECH Data Services	7
2.4	Configuring Messaging: SMS	8
2.5	Configuring Messaging: IVR	8
2.6	Connecting MOTECH to OpenMRS	8
2.7	Integrating MOTECH with CommCare	8
2.8	Using the Tasks Module	8
2.9	Configuring Pill Reminders	9
2.10	Configuring Your MOTECH App to be HIPAA Compliant	9
2.11	MOTECH Mailing Lists	9
3	Architecture and Technical Overviews	11
3.1	Core Architecture	11
3.2	Event and Scheduler Architecture	13
3.3	Modules Architecture	16
3.4	Data Services Architecture	16
3.5	Security Model	16
4	Modules	17
4.1	Alerts	17
4.2	Appointments	17
4.3	Call Flow	17
4.4	CMS Lite	17
4.5	CommCare	17
4.6	Data Services	17
4.7	Decision Tree[Deprecated]	17
4.8	Email	18
4.9	Event Aggregation	18
4.10	Event Logging	18
4.11	IVR	18
4.12	IVR Asterisk	18
4.13	IVR Kookoo	18

4.14	IVR Verboice	18
4.15	IVR Voxeo	18
4.16	Message Campaign	18
4.17	Metrics	18
4.18	Mobile Forms	19
4.19	MRS (Medical Record System)	19
4.20	OpenMRS	19
4.21	Outbox	19
4.22	Pill Reminder	19
4.23	Schedule Tracking	19
4.24	Scheduler	19
4.25	SMS	19
4.26	Tasks	19
5	Developing the MOTECH Platform	21
5.1	Setting up a Development Environment	21
5.2	Developing and Submitting a Patch	35
5.3	MOTECH Code Repositories	36
5.4	Commit Message Format	37
5.5	Coding Conventions	38
5.6	Code Review Workflow and Checklist	38
5.7	Authoring Documentation	38
5.8	Project Management	39
5.9	Release Strategy and Procedure	39
5.10	Browser Compatibility	39
6	Demos	41
6.1	Demo: Create a Message Campaign	41
6.2	Demo: Create a Care Schedule	41
6.3	Demo: Modeling a New System with Data Services	41
6.4	Demo: Hello World	41
6.5	Demo: OpenMRS Schedule Tracking	41
7	Javadoc	43
7.1	org.motechproject.admin.domain	43
7.2	org.motechproject.admin.messages	48
7.3	org.motechproject.admin.service	50
7.4	org.motechproject.bundle.extender	52
7.5	org.motechproject.commons.api	54
7.6	org.motechproject.commons.api.json	62
7.7	org.motechproject.commons.api.model	63
7.8	org.motechproject.commons.couchdb.dao	63
7.9	org.motechproject.commons.couchdb.lucene.query	66
7.10	org.motechproject.commons.couchdb.model	67
7.11	org.motechproject.commons.couchdb.query	67
7.12	org.motechproject.commons.couchdb.service	68
7.13	org.motechproject.commons.date	69
7.14	org.motechproject.commons.date.model	69
7.15	org.motechproject.commons.date.util	72
7.16	org.motechproject.commons.date.util.datetime	77
7.17	org.motechproject.commons.date.valueobjects	78
7.18	org.motechproject.commons.sql.service	79
7.19	org.motechproject.config.core	80
7.20	org.motechproject.config.core.constants	80

7.21	org.motechproject.config.core.domain	83
7.22	org.motechproject.config.core.filestore	90
7.23	org.motechproject.config.core.filters	91
7.24	org.motechproject.config.core.service	91
7.25	org.motechproject.config.domain	92
7.26	org.motechproject.config.monitor	95
7.27	org.motechproject.config.service	96
7.28	org.motechproject.email.builder	103
7.29	org.motechproject.email.contract	106
7.30	org.motechproject.email.domain	108
7.31	org.motechproject.email.service	113
7.32	org.motechproject.event	116
7.33	org.motechproject.event.listener	119
7.34	org.motechproject.event.listener.annotations	121
7.35	org.motechproject.event.queue	124
7.36	org.motechproject.mds.annotations	125
7.37	org.motechproject.mds.builder	127
7.38	org.motechproject.mds.config	131
7.39	org.motechproject.mds.domain	137
7.40	org.motechproject.mds.dto	176
7.41	org.motechproject.mds.enhancer	211
7.42	org.motechproject.mds.ex	212
7.43	org.motechproject.mds.filter	221
7.44	org.motechproject.mds.javassist	223
7.45	org.motechproject.mds.jdo	229
7.46	org.motechproject.mds.query	236
7.47	org.motechproject.mds.repository	244
7.48	org.motechproject.mds.service	252
7.49	org.motechproject.mds.service.impl	267
7.50	org.motechproject.mds.util	277
7.51	org.motechproject.osgi.web	300
7.52	org.motechproject.osgi.web.domain	318
7.53	org.motechproject.osgi.web.exception	319
7.54	org.motechproject.osgi.web.ext	320
7.55	org.motechproject.osgi.web.service	323
7.56	org.motechproject.osgi.web.settings	324
7.57	org.motechproject.osgi.web.util	325
7.58	org.motechproject.scheduler.builder	327
7.59	org.motechproject.scheduler.contract	328
7.60	org.motechproject.scheduler.exception	340
7.61	org.motechproject.scheduler.factory	341
7.62	org.motechproject.scheduler.service	341
7.63	org.motechproject.scheduler.service.impl	346
7.64	org.motechproject.security.annotations	350
7.65	org.motechproject.security.authentication	351
7.66	org.motechproject.security.builder	354
7.67	org.motechproject.security.constants	355
7.68	org.motechproject.security.domain	358
7.69	org.motechproject.security.ex	368
7.70	org.motechproject.security.model	370
7.71	org.motechproject.security.repository	377
7.72	org.motechproject.security.service	383
7.73	org.motechproject.server.api	390
7.74	org.motechproject.server.config	396

7.75	org.motechproject.server.config.domain	399
7.76	org.motechproject.server.config.service	406
7.77	org.motechproject.server.startup	408
7.78	org.motechproject.server.ui	409
7.79	org.motechproject.server.ui.ex	410
7.80	org.motechproject.server.web.controller	410
7.81	org.motechproject.server.web.form	415
7.82	org.motechproject.server.web.helper	421
7.83	org.motechproject.server.web.validator	423
7.84	org.motechproject.tasks.annotations	427
7.85	org.motechproject.tasks.contract	428
7.86	org.motechproject.tasks.domain	435
7.87	org.motechproject.tasks.ex	479
7.88	org.motechproject.tasks.json	480
7.89	org.motechproject.tasks.repository	482
7.90	org.motechproject.tasks.service	482
7.91	org.motechproject.tasks.util	492
8	Contribute	495
8.1	Development	495
8.2	Documentation	496
8.3	Translation	496
9	Indices and tables	497

The topics below will give you an introduction to MOTECH, an open source mHealth platform developed by the Grameen Foundation. Some of these topics are quite technical in nature (i.e. aimed at software developers), while others are more accessible to people who aren't familiar with software code.

About MOTECH

Mobile Technology for Community Health (MOTECH) is a modular, extensible open source software project originally designed for mobile health (mHealth), which can also be used outside of the health domain. It allows organizations to use mobile technology to communicate information to patients, collect data about patients, alert caregivers to patients' status, and schedule caregivers' work. The modular system allows organizations to choose among multiple mHealth technologies and to enable data sharing for users of the system.

1.1 MOTECH Architecture Overview

MOTECH consists of the core platform and several modules, each providing use of a technology such as SMS or email, or access to an external system such as CommCare. Organizations can choose to install one or more modules, and developers can extend MOTECH by writing new modules. MOTECH is written in Java. It depends on open source systems including Apache Tomcat, Apache ActiveMQ, and Quartz. For more information about MOTECH architecture, see *Core Architecture* and *Modules Architecture*.

1.2 What can MOTECH do?

The MOTECH Platform can be used for setting appointments, tracking any scheduled activity, and managing workers deployed in the field. Its initial implementations have been for mHealth projects that improve health by sending messages to patients and caregivers based on an evaluation of the recommended schedule of care compared to the patient's health-related actions. Some features of typical MOTECH-based applications are:

Communicate information to patients via voice or SMS according to a schedule of care defined for the patient's condition, e.g.:

- Reminders for ANC appointments, lab visits, etc.
- Reminders to take medication on schedule, e.g., DOTS, ART, etc.
- Reminder notices to take children for scheduled immunization services

Collect data from patients or caregivers, e.g.:

- Patients report their symptoms prior to treatment or during treatment (adverse events)
- Patients give feedback on service delivery
- Caregivers report what service was delivered to a patient and on what date

Alert caregivers of the status of their patients, e.g.:

- Notify Community Health Worker when patient has not taken ART, DOTS or other drugs

- Notify nurse when patient has not kept a scheduled appointment (e.g., ANC visit)

Facilitate communication between patients, caregivers, and/or health administrators, e.g.:

- Establish secure peer networks for patients who share similar health concerns
- Initiate conversations between patients and caregivers in a way that allows the caregiver to manage the workload effectively

Creating Applications

This section is intended to help implementers of MOTECH get started creating custom solutions. The topics below cover some of the common features that need to be configured for a MOTECH app.

As the platform matures, most of the features below will be usable by implementers without developing any software code. As of now, a number of these features do require some coding in order to use (the topics below provide sample code where appropriate).

2.1 Introduction

Motech provides [Maven archetypes](#) in its [Nexus repository](#) which you can use to create a new Motech module. The archetypes supply basic source code needed for a new module, as well as configuration files for packaging the module as a bundle to be loaded into a Motech server.

The first archetype is the *minimal bundle archetype*. This supplies just enough source code and configuration to make a “Hello World” module.

Additional archetypes can add functionality to the minimal archetype:

- The *http bundle archetype* adds a servlet to respond to HTTP requests, and a simple web user interface.
- The *repository bundle archetype* adds a repository layer for storing and retrieving data from *MOTECH's data services*.
- The *settings bundle archetype* adds a properties file to store custom module configuration, and exposes the configuration through Motech's web user interface

Any combination of these additional archetypes may be added to the minimal archetype.

2.1.1 Minimal Bundle Archetype

To create a new minimal bundle from the minimal bundle archetype, use the following command:

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeRepository=http://nexus.motechproject.org/
```

This will create a new Maven project in your current directory.

This is a long command. Here is an explanation of the parameters:

<i>parameter</i>	<i>value</i>	<i>explanation</i>
-DinteractiveMode	false	no need to wait for user input
-DarchetypeRepository	http://nexus.motechproject.org/content/repositories/releases	where to find the archetype
-DarchetypeGroupId	org.motechproject	group name for the archetype
-DarchetypeArtifactId	minimal-bundle-archetype	which archetype to use
-DarchetypeVersion	0.24-SNAPSHOT	Motech version to use with the new module
-DgroupId	org.motechproject	group name for the new module
-DartifactId	motech-test-module	artifact name for the new module
-Dpackage	archetype.test	Java package for new module classes
-Dversion	0.1-SNAPSHOT	version of the new module itself
-DbundleName	"Archetype Test Module"	name of the new module

2.1.2 HTTP Bundle Archetype

To create a new bundle that has HTTP support, use the following two commands from the same directory.

Create a minimal bundle with configuration modified for HTTP:

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeRepository=http://nexus.motechproject.org/
```

Note the new parameter:

-Dhttp	true
--------	------

Add new source files from the HTTP archetype:

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeRepository=http://nexus.motechproject.org/
```

Note the new archetype Id:

-DarchetypeArtifactId	http-bundle-archetype
-----------------------	-----------------------

2.1.3 Repository Bundle Archetype

To create a new bundle that has support for *MOTeCH's data services*, use the following two commands from the same directory.

Create a minimal bundle with configuration modified for repository:

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeRepository=http://nexus.motechproject.org/
```

Add new source files from the repository archetype:

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeRepository=http://nexus.motechproject.org/
```

2.1.4 Settings Bundle Archetype

To create a new bundle that has module settings support, use the following two commands from the same directory.

Create a minimal bundle with configuration modified for settings:

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeRepository=http://nexus.motechproject.org/
```

Add new source files from the settings archetype:

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeRepository=http://nexus.motechproject.org/
```

2.1.5 Combined Bundle Archetype

The minimal bundle archetype can be supplemented with any combination of additional archetypes. To create a bundle that uses them all, use all the following commands from the same directory.

Create a minimal bundle with configuration modified for all additional archetypes:

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeRepository=http://nexus.motechproject.org/
```

Add source files from all the additional archetypes:

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeRepository=http://nexus.motechproject.org/
```

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeRepository=http://nexus.motechproject.org/
```

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeRepository=http://nexus.motechproject.org/
```

2.1.6 Using Archetypes Locally

You can also use the archetypes locally, without the Motech Nexus repository. First, you must build the archetypes locally. You can either follow the `developer` guidelines to set up your developer environment, or to build locally without committing:

```
git clone https://code.google.com/p/motech/
cd motech
mvn clean install
```

Then you can use the archetypes from your Maven local catalog:

```
mvn archetype:generate -DinteractiveMode=false -DarchetypeCatalog=local -DarchetypeGroupId=org.motech
```

Note the new parameter:

-DarchetypeCatalog	local
--------------------	-------

2.2 Installing and Upgrading MOTECH

There will be more text here.

2.3 Modeling Data with MOTECH Data Services

There will be more text here.

2.3.1 Javadoc

org.motechproject.mds.service
org.motechproject.mds.annotations
org.motechproject.mds.builder
org.motechproject.mds.config
org.motechproject.mds.domain
org.motechproject.mds.dto
org.motechproject.mds.enhancer
org.motechproject.mds.ex
org.motechproject.mds.filter
org.motechproject.mds.jdo
org.motechproject.mds.repository
org.motechproject.mds.util
/org/motechproject/mds/web/package-index

2.4 Configuring Messaging: SMS

There will be more text here.

2.5 Configuring Messaging: IVR

There will be more text here.

2.6 Connecting MOTECH to OpenMRS

There will be more text here.

2.7 Integrating MOTECH with CommCare

There will be more text here.

2.8 Using the Tasks Module

There will be more text here.

2.9 Configuring Pill Reminders

There will be more text here.

2.10 Configuring Your MOTECH App to be HIPAA Compliant

There will be more text here.

2.11 MOTECH Mailing Lists

The mailing lists below are the best way to keep in touch with us. Please join the discussion!

2.11.1 MOTECH Developers

Mailing list for regular contributors to the MOTECH Platform source repository - used for design discussions and other issues impacting the developer community.

motech-dev@googlegroups.com	Join Dev List
--	-------------------------------

2.11.2 MOTECH Implementers

Mailing list for implementers and other users of MOTECH - used mostly for communication related to releases. Traffic is moderated and very low volume.

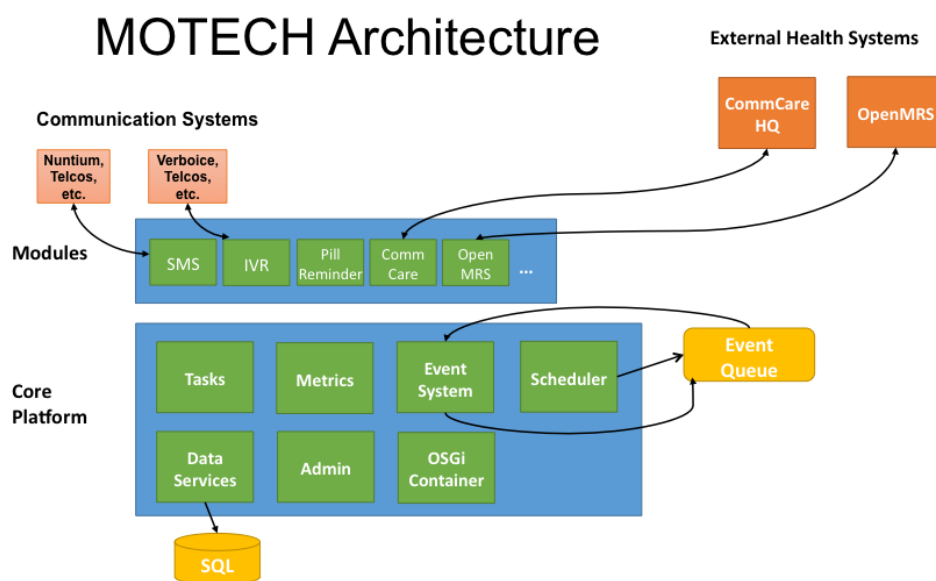
motech-implementers@googlegroups.com	Join Implementers List
--	--

Architecture and Technical Overviews

3.1 Core Architecture

3.1.1 Architecture Overview

MOTECH can logically be broken into the core platform and modules. The core platform wraps several well-known open source systems, and augments and exposes their features to the other components. The main functions of the core are to wrap ActiveMQ (which provides the message queue) and present an internal pub/sub like event interface to the module and implementation layers. The core also provides a module loading environment (OSGi), an interface to the Scheduler, and access to the database.



Modules within MOTECH are self-contained bits of functionality that are loaded into the server via the OSGi host. Typically a module provides one type of functionality, such as SMS or access to an external health system. For more information, see [Modules Architecture](#). For a list of current and planned modules, see [Modules](#).

MOTECH is designed to be horizontally scalable with multiple MOTECHs all acting as workers connecting to the same message queue.

3.1.2 Design Philosophy

Stateless

A core design principle of the MOTECH platform is that the server should be stateless across requests to allow for horizontal scalability. It is expected that code running within the MOTECH server should perform a single action per request and then return. The module should never persist any state in memory or local disk and expect that state to be available to later requests.

Events

To aid in the development of stateless services, the MOTECH engine provides a pub/sub like event system. (The event system follows the publish-subscribe pattern but does not implement the standard Java pub/sub protocol.) It helps to decouple emitters of events from the modules that wish to consume them. Any module can emit an event by calling the `EventRelay` and passing it a `MotechEvent` and a subject. To register for an event, a module just needs to annotate a method with the list of event subjects of interest.

For more information, see [Event and Scheduler Architecture](#).

Scheduled Events & Timers

To assist in the development of a stateless event-based server, the MOTECH platform provides access to a flexible scheduling system. Using the open source Quartz engine, MOTECH can easily schedule events for future consumption. For more information, see [Event and Scheduler Architecture](#).

3.1.3 Subsystems

Tasks System

The Tasks system allows you to connect modules without code by using tasks. Each task consists of three parts:

1. Trigger: an event raised by Module A (or the Scheduler)
2. Filter: a conditional statement specifying whether the task should run
3. Action: an action executed by Module B in response

In between the trigger and the action, tasks may use data loaders to look up data from other modules that are registered as data sources.

Data Services

MOTECH Data Services is a flexible data modeling system that allows users to define and share custom schemas without code, and provides auditing and revision tracking. It is a JDBC-based user configurable database abstraction layer on top of a standard SQL database. It provides generated POJOs and OSGi service interfaces for the data objects, generated CRUD events, and generated user interface for data browsing and editing. In a future release it will also support auto-generation of REST APIs.

3.1.4 Dependencies on Third-Party Systems

Quartz Scheduler

Quartz is an open source job scheduling engine that enables MOTECHE modules to schedule events for future consumption.

Tomcat

Apache Tomcat provides the application container for MOTECHE.

ActiveMQ

Apache ActiveMQ is an open source message broker that provides the message queue.

OSGi

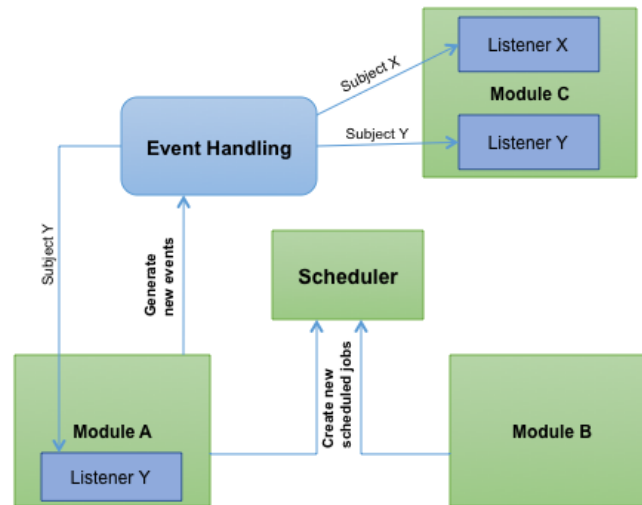
Each MOTECHE module is an OSGi bundle. Using OSGi allows the platform to manage the bundle lifecycle (adding, removing, starting, and stopping modules), and allows modules to expose service interfaces. For more information, see *Modules Architecture*.

3.2 Event and Scheduler Architecture

3.2.1 Event Handling and Scheduling among Modules

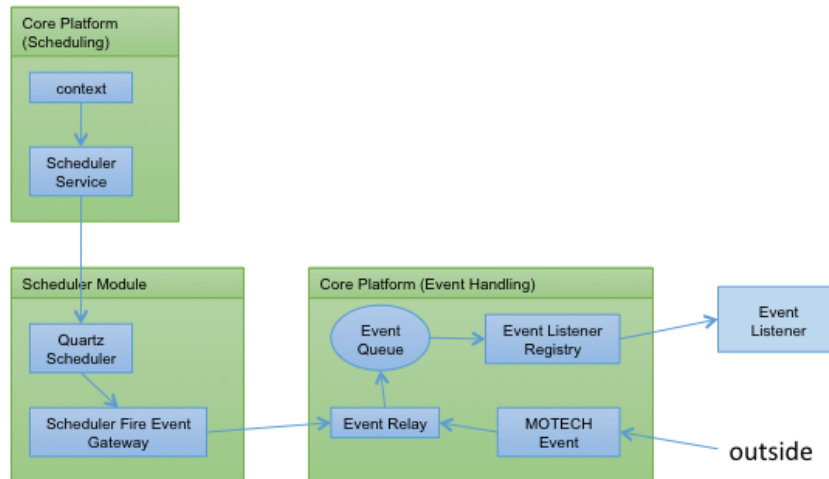
The following diagram provides three examples of Motech modules:

- Motech module A publishes events, schedules new jobs and listens for events. An example of a Motech platform that has these three responsibilities is the Message Campaign module.
- Motech module B schedules new jobs.
- Motech module C listens to events of subject X and Y: listener X listens to events of subject X, and listener Y listens to events of subject Y.



3.2.2 Event Handling and Scheduling Architecture

To schedule a job in MOTECH, the core platform exposes the `MotechSchedulerService`. Clients of the service have an instance of it injected into the class that uses it. This service employs Quartz to schedule `MotechScheduledJobs`. When triggered, `MotechScheduledJobs` raise events that are sent through an event relay to the event queue. These messages are dequeued and then received by a consumer, the event listener registry, which in turn discovers all of the listeners on the event and invokes the appropriate method that was listening on that event.



Notes

- The scheduler service retrieves the Quartz Scheduler from the SchedulerFactoryBean. The service can only schedule MotechScheduledJobs, which all must include a MotechEvent.
- When the trigger for the scheduled job is satisfied, the job is executed and the accompanying Motech event is sent to the SchedulerFireEventGateway interface. This gateway is defined in schedulerFiredEventChannelAdapter.xml and a proxy is generated by Spring at runtime.
- The SchedulerFireEventGateway serves as an outbound message gateway to shuttle messages to the event relay, which then sends JMS messages to the event queue.
- The event queue dequeues messages and routes them to the event listener registry. The core platform has one consumer for events, a channel that routes messages to an event listener registry.
- The event listener registry retrieves the listeners that are listening on the motech event it is relaying (listening is based on the value bound to the motech event's subject key).
- The core platform scans new bundles for any method annotated as a MotechListener, and registers those methods with the listener registry.
- If the event has a specific destination or only one listener, the listener's handle method is called. This will invoke the method that was annotated as a MotechListener. These listener classes can be project specific and will reside outside of the core platform.
- If the event has multiple listeners, a separate event is raised for each listener. The original event object is not handled by a listener in this case.
- The EventListener's handle method will invoke the method that was annotated as a MotechListener. The MotechEvent will be passed to that method as a parameter.

3.3 Modules Architecture

Modules within MOTECH are self-contained bundles of functionality that are loaded into the server via the OSGi host. Modules interact with the core platform through its APIs and with other modules, either through their APIs or by consuming their events. Modules can expose service interfaces of their own as well as emit their own events. Modules may also register servlet controllers, which allow them to respond to HTTP requests.

Through MOTECH Data Services, a modules may expose entities from its data model. This allows a module to provide a data editor, REST APIs, record-level security, and field-level auditing. Via the Tasks system, modules can expose triggers, data and actions to be orchestrated by other modules.

See [Modules](#) for the list of current and planned modules.

Reasons to create a module include developing application-specific UI, business logic, and data models. Another reason is to develop generic reusable functionality to share with the MOTECH community.

3.4 Data Services Architecture

There will be text here.

3.5 Security Model

There will be text here.

4.1 Alerts

Collects alerts for users in an inbox-like container

4.2 Appointments

Provides appointment scheduling and reminding

4.3 Call Flow

Manages the sequence of greetings and prompts for an Interactive Voice Response (IVR) call

4.4 CMS Lite

Provides basic content storage and retrieval

4.5 CommCare

Integrates the MOTECH platform with CommCareHQ, an open-source platform to help manage community health workers

4.6 Data Services

Integrates data from external data sources and provides sharable data schemas

4.7 Decision Tree[Deprecated]

Provides APIs for constructing an IVR decision tree

4.8 Email

Sends and logs email messages

4.9 Event Aggregation

Groups common events and republishes them as a single event at a specified time

4.10 Event Logging

Allows MOTECH modules to easily see each others' events

4.11 IVR

Integrates the MOTECH platform with Interactive Voice Response (IVR) services

4.12 IVR Asterisk

Connects the MOTECH platform IVR with an Asterisk server using the VoiceGlue VoiceXML browser

4.13 IVR Kookoo

Integrates the MOTECH platform with Kookoo's hosted IVR service

4.14 IVR Verboice

Integrates the MOTECH platform with Verboice's hosted IVR service

4.15 IVR Voxeo

Integrates the MOTECH platform with Voxeo's hosted IVR service

4.16 Message Campaign

Enrolls users in message campaigns with flexible content-scheduling rules

4.17 Metrics

Displays web site metrics using logging and the open source tools StatsD and Graphite

4.18 Mobile Forms

Supports configurable forms and data collection through mobile devices that support the OpenXData format

4.19 MRS (Medical Record System)

Provides a basic specification for integrating the platform with a medical record system

4.20 OpenMRS

Integrates the MOTECH platform with OpenMRS, an open source electronic medical record platform

4.21 Outbox

A voicemail-like messaging system for end users

4.22 Pill Reminder

A flexible reminder system focused on medication

4.23 Schedule Tracking

Enrolls users for alerts based on complex scheduling rules

4.24 Scheduler

Publishes events on a schedule, using the open source Quartz engine.

4.25 SMS

Provides a basic specification for integrating the MOTECH platform with an SMS provider to send/receive SMS messages

4.26 Tasks

Responds to specified triggers; for example, a task can be set to enroll a patient in an alerts schedule in response to an incoming SMS message with a particular subject

Developing the MOTECH Platform

This section of the documentation is aimed at developers and maintainers of the MOTECH Platform. These docs help MOTECH devs get up and running, from configuring a machine, to getting the code, and committing and reviewing changes. If you find any errors in the topics below, or you have any other questions about MOTECH development, please contact us via the [mailing list](#).

5.1 Setting up a Development Environment

The topics below will walk you through installing MOTECH on a Mac or Linux machine. Note that the Docker-based setup is much faster than the “official” method but the instructions are in beta. If you have any trouble with either approach, please feel free to contact us via motech-dev@googlegroups.com.

5.1.1 Installing MOTECH for Developers (“Official” Method)

Table of Contents

- Installing MOTECH for Developers (“Official” Method)
 - Installing on Ubuntu
 - Installing on a Macintosh
 - Building and Installing MOTECH
 - Installing the IDE, IntelliJ IDEA Community Edition & open MOTECH project

Installing on Ubuntu

The versions below may change, most likely the latest stable release will work for your purposes. If they do not, please feel free to send in feedback.

1. **Install Ubuntu Desktop 12.04.2 LTS 64bit** [Installation instructions](#)

Note: 64-bit is required for Motech’s installation

2. Install Maven, Git, Curl, Activemq, and mysql

(a) In terminal, type

```
sudo apt-get install curl git maven activemq mysql-server
```

- (b) On a fresh Ubuntu installation, you may need to run the following first

```
sudo apt-get update
```

3. Configure ActiveMQ

Run the following

```
sudo ln -s /etc/activemq/instances-available/main /etc/activemq/instances-enabled/main
```

Note: For ActiveMQ scheduled delivery to work, you must set the attribute: **schedulerSupport="true"** for the broker element in your activemq.xml config file. This file should be located at (active-mq-folder)/conf/activemq.xml. See ActiveMQ docs.

4. Install JDK 7

- (a) Go to [The Java JDK Download Page](#)
- (b) Accept License Agreement
- (c) Click on `jdk-7u51-linux-x64.tar.gz` (or latest stable version)
- (d) Extract the file into your home directory, ie: `/home/*<user>*/jdk1.7.0_51`
- (e) Set the proper Java environment and change maven options:

i. Start a new terminal session

ii. Edit your `.profile` file

```
nano ~/.profile
```

iii. append the following at the end of the file:

```
export PATH="$HOME/jdk1.7.0_21/bin:$PATH"
export JAVA_HOME=$HOME/jdk1.7.0_21
export MAVEN_OPTS="-Xmx512m -XX:MaxPermSize=128m"
```

iv. Save the changes (Ctrl+X) and quit

v. Confirm the settings are right

vi. Log out & log back in & start a new terminal

vii. Type

```
java -version && env | grep MAVEN_OPTS
```

You should see something like:

```
java version "1.7.0_51"
Java(TM) SE Runtime Environment (build 1.7.0_51-b11)
Java HotSpot(TM) 64-Bit Server VM (build 23.21-b01, mixed mode)
MAVEN_OPTS=-Xmx512m -XX:MaxPermSize=128m
```

5. Install Tomcat7

- (a) Go to [Tomcat's download page](#)

- (b) Under 7.0.52 (or the latest stable version) - Binary Distributions - Core, click on tar.gz
- (c) Once downloaded, expand the file to your home directory, i.e.:
`/home/*<user>*/apache-tomcat-7.0.52`
- (d) Edit the `tomcat-users.xml` file (located under `\etc\tomcat7\conf\`) to add an admin user:
- (e) In the terminal type

```
nano ~/apache-tomcat-7.0.52/conf/tomcat-users.xml
```

- (f) Insert a line similar to the following before the closing `</tomcat-users>` tag:

```
<user username="*<username>*" password="*<password>*" roles="manager-gui"/>
```

- (g) Save the changes (Ctrl+X) then quit
- (h) Now edit `~/ .bashrc` to setup tomcat's environment variable

```
nano ~/ .bashrc
```

- (i) Append the following line:

```
export CATALINA_HOME=$HOME/apache-tomcat-7.0.52
```

- (j) Save the changes (Ctrl+X) then quit
- (k) Start a new terminal session or type

```
source ~/ .bashrc
```

6. Install CouchDB

- (a) Download the latest stable sources from <http://couchdb.apache.org> and then follow the instructions in the `INSTALL.Unix` file.

For Ubuntu 13.10, just run:

```
sudo apt-get install couchdb
```

- (b) Once this is done, navigate to <http://localhost:5984> to verify that the installation completed successfully.

```
{ "couchdb": "Welcome", "uuid": "52068def93b82a2653dcf352a4f9273a", "version": "1.4.0", "vendor": {
```

7. Install CouchDB-Lucene

- (a) Follow [these instructions](#) also be sure to follow the [proxy handler instructions](#)

- (b) Once the proxy has been configured, restart couchdb with:

```
sudo service couchdb restart
```

- (c) After restarting couchdb, navigate to http://localhost:5984/_fti and you should see something like this:

```
{ "couchdb-lucene": "Welcome", "version": "0.10.0-SNAPSHOT" }
```

8. Setup MySQL

- (a) In your motech source root directory, type in the terminal:

```
$ mysql -u root -p
```

(b) then type:

```
sql> create database motechquartz;
sql> create database motech_data_services;
sql> create user 'quartz'@'localhost' identified by 'quartz2123';
sql> grant all privileges on motechquartz.* to 'quartz'@'localhost';
sql> exit;
```

(c) then type:

```
mysql -u root -p motechquartz < modules/scheduler/scheduler/sql/create_db_schema_quartz_
```

9. Start Tomcat

(a) In terminal, type:

```
~/apache-tomcat-7.0.52/bin/catalina.sh jpda start
```

(b) You should see messages similar to:

```
Using CATALINA_BASE:   /home/*<user>*/apache-tomcat-7.0.52
Using CATALINA_HOME:   /home/*<user>*/apache-tomcat-7.0.52
Using CATALINA_TMPDIR: /home/*<user>*/apache-tomcat-7.0.52/temp
Using JRE_HOME:        /home/*<user>*/jdk1.7.0_51
Using CLASSPATH:       /home/*<user>*/apache-tomcat-7.0.52/bin/bootstrap.jar:/home/*<use
```

(c) You can also confirm tomcat was started by going to <http://localhost:8080> in a browser

10. Jump to the [Building and Installing MOTECH](#) section to install MOTECH

Installing on a Macintosh

1. Installing Prerequisites for MOTECH

(a) Installing [HomeBrew](#)

To install Homebrew, run the following in the terminal

```
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/homebrew/go/install)"
```

(b) Use Homebrew to install git, erlang, ActiveMQ, and Apache Tomcat:

```
brew install git
brew install activemq
brew install tomcat
brew install maven
```

(c) Homebrew installations are located in `/usr/local/Cellar`` with symlinks in ```/usr/local/bin`, which should already be part of your `$PATH` environment variable.

Note: Homebrew provides instructions about how to run these applications, as well as how to have launchd start them automatically on system startup.

(d) Configuring Tomcat:

Edit the `tomcat-users.xml` file to add an admin user. Insert a line similar to the following before the closing `</tomcat-users>` tag:

```
<user username="motech" password="motech" roles="manager-gui"/>
```

(e) Installing JDK 7:

Mac OS includes JDK6 by default, however JDK 7 is required for MOTECH. Use these [instructions](#) to install the latest version of the JDK.

(f) Installing MySQL:

- i. Before installing MySQL, you will need Xcode from the App Store. This can take a while; it's a big download.
- ii. Next start Xcode from the Launchpad (rocketship icon in the dock) and select Install. Then you can quit Xcode; you don't need to keep it running.

Note: (Command Line Tools using Xcode are included in OS X Mavericks, but not previous OS versions. If you are running Mountain Lion, you can follow these instructions: <http://blog.mclaughlinsoftware.com/2012/12/10/mountain-lion-pre-mysql/>)

- iii. Go to <http://dev.mysql.com/downloads/mysql/> and download the appropriate DMG archive. Open it, double-click on the installer, and follow directions.
- iv. Once mysql has finished installing, double-click the MySQL preferences pane in the DMG and follow instructions. For more details see 'these instructions' _ .

Note: Homebrew can be used to install MySQL, however Homebrew will not install the Mysql System Preferences control panel.

2. Setting up Symbolic Link and Environment Variables

- (a) Create a symbolic link from the Tomcat directory (Homebrew installs into `/usr/local/Cellar/tomcat/<version number>/libexec`) to `/usr/local/tomcat`:

```
ln -s /usr/local/Cellar/tomcat/`brew info tomcat | grep stable | awk '{print $3}' | sed
```

- (b) Edit your `~/.bash_profile` to set environment variables (catalina is Tomcat):

```
export JAVA_HOME="/Library/Java/Home"
export MAVEN_OPTS="-Xmx512m -XX:MaxPermSize=128m"
export CATALINA_HOME="/usr/local/tomcat"
export CATALINA_OPTS="-Xms1024m -Xmx2048m -XX:MaxPermSize=1024m"
export PATH=/usr/local/mysql/bin:$PATH
```

- (c) **When you're done editing:**

```
source ~/.bash_profile
```

3. Jump to the [Building and Installing MOTECH](#) section to install MOTECH

Building and Installing MOTECH

1. Getting the MOTECH code

[List of MOTECH repositories](#)

[Generic developer git workflow](#)

2. Building MOTECH

(a) Assuming you issued the git clone command in your home directory root, in the terminal

```
$ cd ~/motech
$ mvn install
```

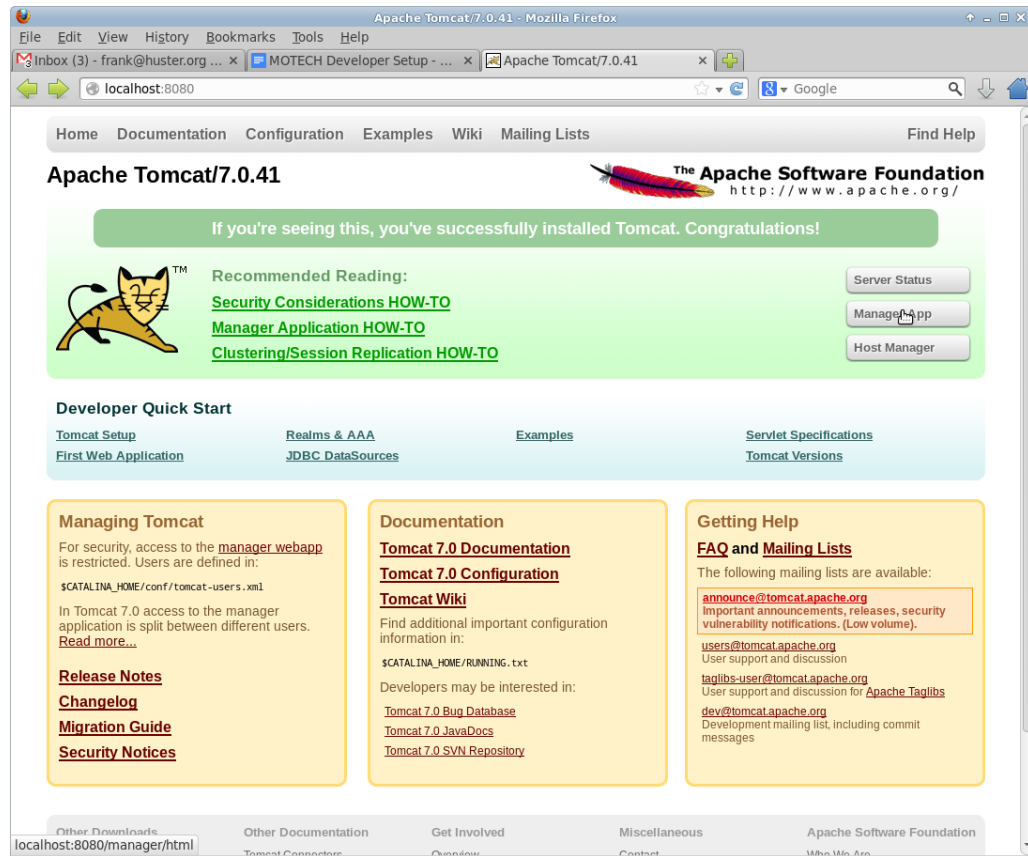
b.) It takes some time to build MOTECH, but eventually you should see:

```
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 29:19.284s
[INFO] Finished at: Fri Jun 07 12:12:43 PDT 2013
[INFO] Final Memory: 152M/378M
[INFO] -----
```

Note: Should you get a `java.lang.OutOfMemoryError` exception, it may be because you forgot to set `MAVEN_OPT` as described in [3.5]. But you may need to increase `-Xmx`. So something like `-Xmx1024m` might work.

3. Install MOTECH

(a) In a browser, go to <http://localhost:8080>



(b) Click on Manager App

(c) Type the user/password you used in tomcat-users.xml

temporary hack you need to remove `~/motech/config/motech-settings.conf` to allow the create initial user wizard.

(d) In the Tomcat Web Application Manager, scroll down to the Deploy section and the WAR file to deploy subsection, click on Browse and select or navigate to `~/motech/platform/server/target/motech-platform-server.war` then click on Deploy

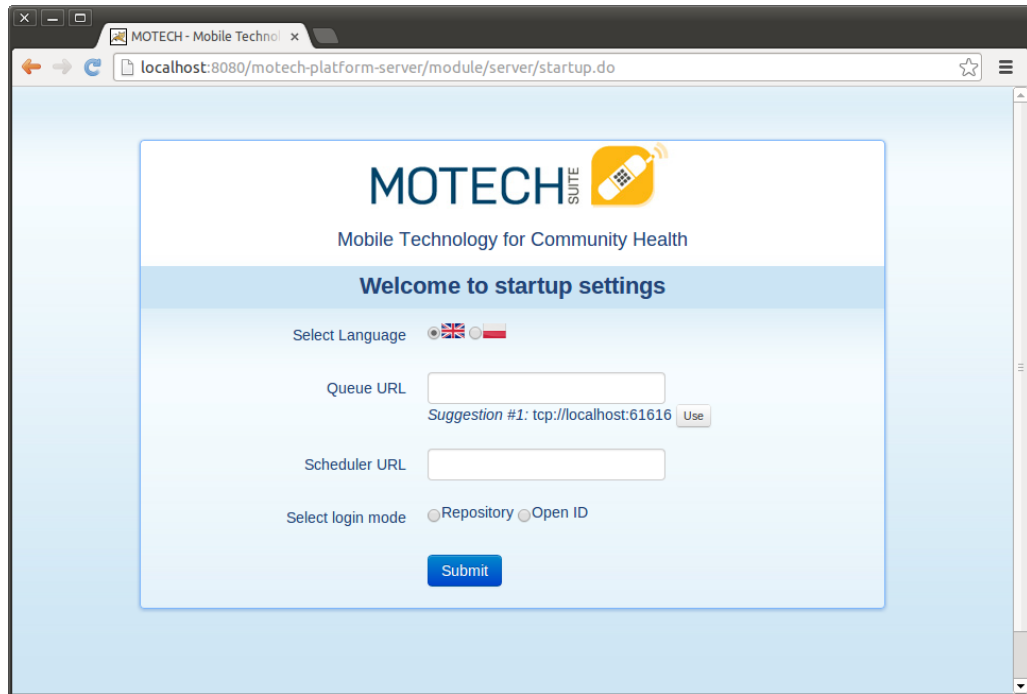
The screenshot shows the Tomcat Web Application Manager interface. The browser address bar indicates the URL is `localhost:8080/manager/html/upload?org.apache.catalina.filters.CSRF_NONCE=D07AE837173CA6D1B14C5F0B4A479366`. The page features the Apache Software Foundation logo and the title "Tomcat Web Application Manager". A message box at the top displays "Message: OK". Below this is a navigation bar with links: "List Applications", "HTML Manager Help", "Manager Help", and "Server Status". The main content area is titled "Applications" and contains a table with the following data:

Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	None specified	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	None specified	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/motech-platform-server	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Below the table is a "Deploy" section with the following input fields:

- Deploy directory or WAR file located on server
- Context Path (required):
- XML Configuration file URL:
- WAR or Directory URL:

- (e) Depending on your machine it could take a while for `motech-platform-server` to deploy
- (f) In the Tomcat Web Application Manager page, click on `/motech-platform-server`, you get the MOTECH initial user screen

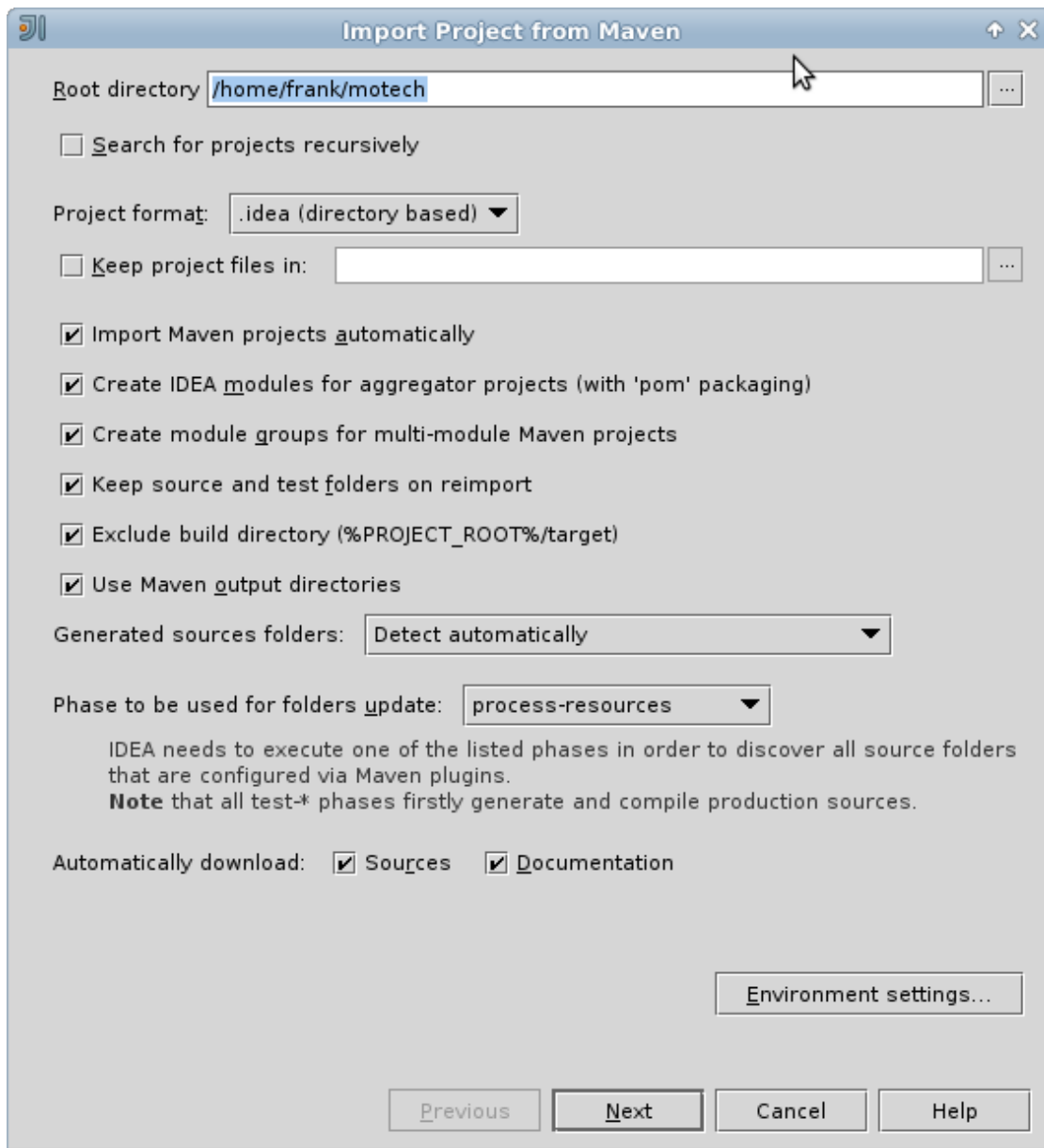


Installing the IDE, IntelliJ IDEA Community Edition & open MOTECH project

1. Go to the [Jetbrains home page](#) and click on Download Now in the Community Edition box, then expand the file to your home directory.
2. From a terminal, assuming you extracted IntelliJ to ~/idea-IC-129.713, start IntelliJ

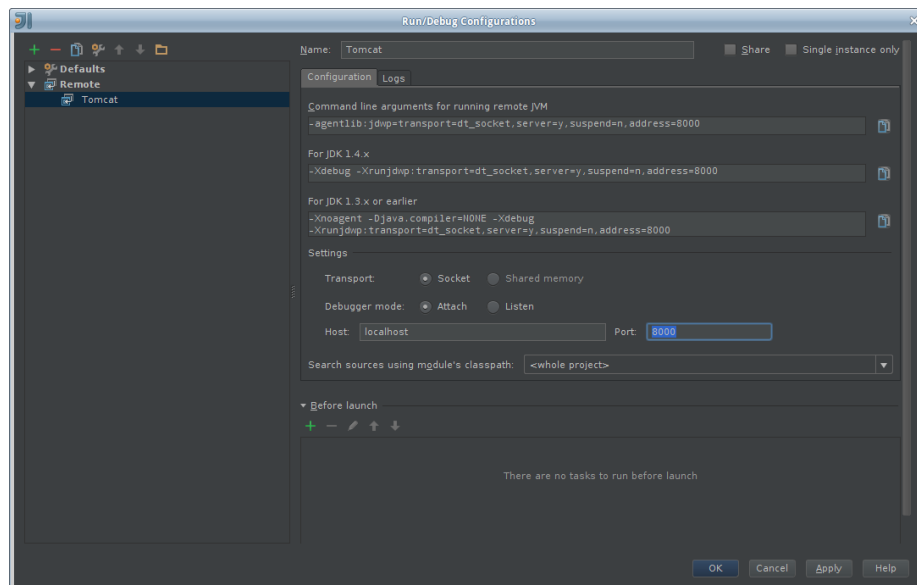
```
$ ~/idea-IC-129.713/bin/idea.sh
```

3. Select Import Project

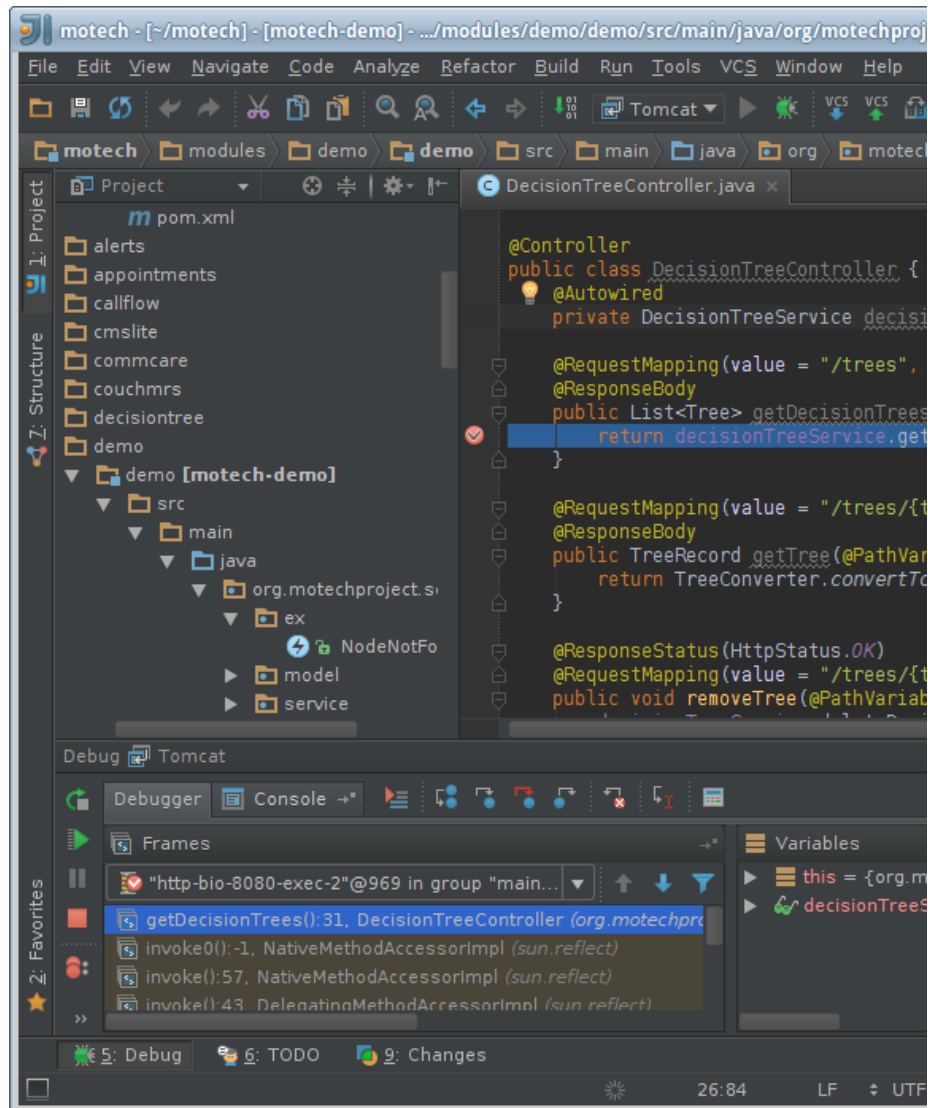


1. Select ~/motech/pom.xml, a dialog box will appear. Set the options as shown:
2. Click Next
3. In Select Profiles, do not select any profile, click Next
4. In Select Maven projects to Import, there should only be one project: org.motechproject:motech:0.20-SNAPSHOT, click Next
5. In Please select project SDK, if the 1.7.0_21 is present, select it, otherwise add it:
6. Click +
7. Select JDK
8. Select /home/frank/jdk1.7.0_21, then click OK
9. Click Next

10. Click Finish
11. Background processes will take a long time
12. You can also create a menu launcher, so you can start IntelliJ from the gui:
 - (a) From the Tools menu select Create Desktop Entry
 - (b) A IntelliJ menu item will be created in the Development application group
 - (c) Debug demo module in IntelliJ
 - (d) Start IntelliJ (from the command line, or from launcher icon if you created one)
 - (e) It'll automatically open the motech project (if it was the last project you worked on)
 - (f) From the Run menu select Edit Configurations
 - (g) Click on the green +
 - (h) Select Remote
 - (i) Give a name to your Run/Debug configuration and change the port to 8000 as:



- (j) Hit OK
- (k) Set a breakpoint somewhere in the demo module code, i.e.:
 - (l) From the Run menu, select Debug 'Tomcat' where Tomcat is the name of your configuration.
- (m) In the browser go to the place that will hit the breakpoint, i.e.: if you setup a breakpoint as in the previous screen, then in the Demo module, click the Decision Trees tab, and you should hit the breakpoint!



5.1.2 Installing MOTECH Using Docker (“Beta”)

Note: These instructions assume you’re running on Ubuntu. This setup is also possible on Mac OSX but the steps are slightly different. We hope to provide OSX instructions in future.

This document provides instructions for creating a MOTECH environment using [Docker](#) containers. These instructions are “in beta” (the *official* installation guide is still the one [here](#)), but many members of the MOTECH dev team have been following this approach with success. This installation method is much faster than the official route.

There are two supported ways to install MOTECH with Docker:

1. As an implementer - follow this approach if you want to install a released version of MOTECH.
2. As a developer - follow this approach if you will be developing MOTECH and want to build the platform and modules from source code.

Get Docker, Fig and motech-docker

Whether you're installing as an implementer or a developer, you'll need Docker and Fig:

Docker

1. Follow the instructions on the [Docker website](#) to get the latest version of Docker.
2. Execute the following to configure Docker to work for non-root users:

```
sudo groupadd docker
sudo gpasswd -a ${USER} docker (logout and re-login)
sudo service docker restart
```

Fig

Execute the following to install Fig:

```
sudo apt-get install python-pip python-dev build-essential
sudo pip install -U fig
```

motech-docker

Clone the [motech-docker](#) project from GitHub or download it as a zip file and extract it. You'll need to run all Fig commands from the motech-docker directory.

Implementer Setup

Go to your motech-docker directory. To setup as an implementer (everything is automagically installed):

```
./setup_as_imp.sh
```

Type the following to start MOTECH in the background:

```
fig up -d
```

Voila! MOTECH has started. Wait a little bit (about 30s) then direct your browser to: <http://localhost:8080/motech-platform-server>

Note: 'fig up' ERASES ALL YOUR DATA (well not really all, but pretend it does). You have to run it at least once to setup MOTECH. If you run it again, it'll erase everything you did in MOTECH. It's useful to start afresh, but remember: it nukes everything!

Developer Setup

Go to your motech-docker directory. To setup as a dev:

```
./setup_as_dev.sh
```

Type the following to start all the pieces that MOTECH needs to run in the background:

```
fig up -d
```

Once you start the containers with the `fig up -d` command above and *before* you build MOTECH for the first time, you need to copy the MOTECH binaries into the container's `/root/.motech/bundles` directory.

Conveniently, the container's `/root/.motech/bundles` directory is exposed as the `docker-motech-bundles` directory (with `a-rw` access) in your home directory (also note that the container's `/root/.motech/config` dir is also exposed as `~/docker-motech-config`). So, you can either manually copy the necessary binaries, or you can create a symbolic link to `~/docker-motech-bundles` from `~/motech/bundles`.

Assuming the latter, and that you never built MOTECH before, you'd run the following commands:

```
# go to your home dir
cd
# create the .motech dir
mkdir .motech
# create the symlink
ln -s ~/docker-motech-bundles .motech/bundles
```

If you built MOTECH before, you can just delete the `bundles` directory and create the symlink using the command above.

Build, deploy and run MOTECH: see `:doc:dev_install`.

Some Useful Fig Commands

Stop MOTECH

```
fig stop
```

Restart MOTECH

```
fig start
```

Watching logs

To watch all the logs (very verbose):

```
fig logs
```

To watch only the tomcat logs:

```
fig logs tomcat
```

See the sections in the generated `fig.yml` to see what other logs you can watch.

5.1.3 Configuring MOTECH

There will be more text here. This doc may belong in “Setting up a Development Environment” or in “Getting Started”.

5.2 Developing and Submitting a Patch

We use a web-based code review system called [Gerrit](#). Using this system, anyone can comment on a proposed change before it is merged to our Git repository. It's pretty easy to use; the instructions below will get you started.

5.2.1 Create a Gerrit Account

1. Navigate to <http://review.motechproject.org/>
2. Click sign in on top-right
3. Select Open ID provider (only Google accounts are enabled)
4. Select user name
5. Upload your SSH public key

5.2.2 Configuring Your Git Client to Use Gerrit

Follow these steps once for each *MOTECH code repository* that you clone.

1. Get source code

```
git clone ssh://<userid>@review.motechproject.org:29418/motech
```

2. Set up review branch

```
cd motech
git config remote.origin.push refs/heads/*:refs/for/*
```

3. Install change-id generation hook

```
scp -p -P 29418 <userid>@review.motechproject.org:hooks/commit-msg .git/hooks/
```

5.2.3 Development Workflow

1. Checkout to feature branch

```
git checkout -b newfeature
```

2. Make changes/test/multiple commits

3. When ready to submit changes: update master, squash commits and merge feature branch

```
git checkout master && git pull --rebase
git merge --squash newfeature
git gui
```

4. Edit commit message using the proper *commit message format*

5. Push changes

```
git push origin
```

5.2.4 Submitting Changes (Patch Set) to Incorporate Review Comments

If you've received some code review feedback and you'd like to make some changes, follow the steps below to add your changes as a new "patch set" to the existing Gerrit code review.

1. **Checkout patch from gerrit change:**
 - (a) Navigate to [http://review.motechproject.org/#/c/<change id>/](http://review.motechproject.org/#/c/<change id>)
 - (b) Copy pull url under patch set section and run
2. Make changes
3. Copy change ID from Gerrit (top section in Gerrit change page)
4. Amend change ID in commit message
5. [Squash commits](#)
6. Push changes

5.2.5 Pushing to Remote Branches (Not for Review)

This practice enables developers to share in-progress feature work with others without actually submitting the changes for review.

1. Use branch namespace dev

```
git checkout -b dev/newfeature
git add . && git commit -m "message"
git push -u origin dev/newfeature:dev/newfeature
```
2. Once done with feature, squash commits and merge with master. Submit for review as mentioned above.

5.2.6 Additional Information

- <http://review.motechproject.org/Documentation/user-upload.html>
- <http://review.motechproject.org/Documentation/user-changeid.html>

5.3 MOTECH Code Repositories

Each repository can be cloned from either GitHub/Google Code or Gerrit. If you're only interested in a copy of the code and will not be contributing, use the GitHub repo (no sign-in required). Otherwise use the Gerrit repo (sign-in required) where each commit will trigger a Jenkins build and be submitted for code review. Jenkins is our continuous integration (CI) system. Once your change is approved and merged by an authorized Gerrit reviewer, it will show up on the GitHub/Google Code repo.

5.3.1 MOTECH Platform

The platform repo contains the motech-platform-server Tomcat servlet. In addition it also contains the essential Admin, Config, Tasks, Motech Data Services, Email, and Scheduler modules.

- Google Code Repository

```
git clone https://code.google.com/p/motech
```

- Gerrit Repository

```
git clone ssh://<userid>@review.motechproject.org:29418/motech
```

5.3.2 MOTECH Modules

This repo is the home of all optional MOTECH modules.

- GitHub Repository

```
git clone https://github.com/motech/modules
```

- Gerrit Repository

```
git clone ssh://<userid>@review.motechproject.org:29418/modules
```

5.4 Commit Message Format

To ensure that our commit messages are both concise and informative, all MOTECH committers are asked to follow the git commit message format outlined below. For reference, Linus Torvalds provides a description of a good commit message [here](#).

5.4.1 Format

1. First line is the Jira issue ID + subject (max 50 chars)
2. Leave a blank line (if not, the entire message is interpreted as subject)
3. Summary (ensure it is wrapped to a reasonable number of characters, e.g. 72, because git log does not handle wrapping).

The description of the change should be both brief and informative. The Linux kernel documentation has this to say about good commit summaries:

...the “summary” must be no more than 70-75 characters, and it must describe both what the patch changes, as well as why the patch might be necessary. It is challenging to be both succinct and descriptive, but that is what a well-written summary should do.

5.4.2 Author/Committer

There is no need to include the names of the developers who developed the change in the subject line. If there is more than one person working on a change, the committer is asked to include the `–author` parameter in his/her git commit to specify the second person’s name.

5.4.3 Example

Here is an example of a MOTECH commit message:

MOTECH-678 Moves campaign modules to github

Disables pillreminder, message-campaign and scheduletracking modules from main pom.xml as they've moved to a new repo on github - <https://github.com/motech/platform-campaigns>

Change-Id: I5964249887160c868fa9598c413aebb93a49fa32

5.5 Coding Conventions

Text to come

5.6 Code Review Workflow and Checklist

Text to come

5.7 Authoring Documentation

This document provides information specific to setting up and authoring Sphinx documentation for MOTECH. The [Sphinx Documentation Guide](#) is also a good resource.

Each MOTECH repository contains a *docs* directory populated with reStructured Text (reST) files. These files are built by Sphinx and hosted at <http://readthedocs.org>. A Sphinx plugin, called Javaspinx, builds Javadoc for the MOTECH codebase.

5.7.1 Installing Sphinx and Javaspinx

If you are working on MOTECH documentation, it is helpful to build the docs locally and review changes in advance of checkin. The steps below will get your environment configured to build Sphinx docs.

Install python utils

```
sudo apt-get install python-setuptools
sudo apt-get install python-dev
sudo apt-get install python-pip
```

Install sphinx

```
sudo easy_install sphinx
```

Install javaspinx and motechjavaspinx

```
sudo apt-get install libxslt-dev
sudo apt-get install libxml2-dev
sudo apt-get install zlib1g-dev
cd docs
sudo pip install -r requirements.txt
```

5.7.2 Building Docs

ReadTheDocs.org automatically builds Sphinx projects, but it is a good idea to build your documents and make sure they render correctly before you push your changes up to the code repository.

To build Sphinx docs, go to the docs directory and type:

```
make html
```

You can then use a web browser to view docs/build/html/index.html and make sure everything looks good.

5.7.3 Authoring and Submitting New Documents

To create a new documentation topic, simply create a file with extension *.rst* under the docs directory. Create a table-of-contents entry for your new doc in the appropriate index.rst file. The [reStructuredText Primer](#) is good to have handy as you write your doc.

When your document is ready for review, follow the instructions for [creating and submitting a patch](#) to submit it for code review.

5.8 Project Management

There will be text here.

5.9 Release Strategy and Procedure

Text to come

5.10 Browser Compatibility

Text to come

Demos

6.1 Demo: Create a Message Campaign

Text to come

6.2 Demo: Create a Care Schedule

Text to come

6.3 Demo: Modeling a New System with Data Services

Text to come

6.4 Demo: Hello World

Text to come

6.5 Demo: OpenMRS Schedule Tracking

Text to come

7.1 org.motechproject.admin.domain

7.1.1 AdminSettings

public class **AdminSettings**

AdminSettings class is used to store settings for Admin module and specifies that these settings are read-only (by checking config source from bootstrap)

Constructors

AdminSettings

public **AdminSettings** (*List<Settings> settingsList*, boolean *readOnly*)

Methods

getSettingsList

public *List<Settings>* **getSettingsList** ()

isReadOnly

public boolean **isReadOnly** ()

setReadOnly

public void **setReadOnly** (boolean *readOnly*)

setSettingsList

public void **setSettingsList** (*List<Settings> settingsList*)

7.1.2 NotificationRule

public class **NotificationRule**

A notification rule persisted in the database. Represents a rule for sending out a single notification. Contains information about this notification's recipient and the `ActionType` representing a method used for notifying the recipient.

Constructors

NotificationRule

public **NotificationRule** ()

NotificationRule

public **NotificationRule** (*String* recipient, *ActionType* actionType, *Level* level, *String* moduleName)

Methods

getActionType

public *ActionType* **getActionType** ()

getId

public *Long* **getId** ()

getLevel

public *Level* **getLevel** ()

getModuleName

public *String* **getModuleName** ()

getRecipient

public *String* **getRecipient** ()

matches

public boolean **matches** (*StatusMessage* message)

setActionType

public void **setActionType** (*ActionType* actionType)

setId

public void **setId** (*Long id*)

setLevel

public void **setLevel** (*Level level*)

setModuleName

public void **setModuleName** (*String moduleName*)

setRecipient

public void **setRecipient** (*String recipient*)

7.1.3 QueueMBean

public class **QueueMBean**

Represents a JMS queue. Holds information about the queue statistics.

Constructors**QueueMBean**

public **QueueMBean** (*String destination*)

Methods**getConsumerCount**

public long **getConsumerCount** ()

getDequeueCount

public long **getDequeueCount** ()

getDestination

public *String* **getDestination** ()

getEnqueueCount

public long **getEnqueueCount** ()

getExpiredCount

public long **getExpiredCount** ()

getInflightCount

public long **getInflightCount** ()

getQueueSize

public long **getQueueSize** ()

setConsumerCount

public void **setConsumerCount** (long *consumerCount*)

setDequeueCount

public void **setDequeueCount** (long *dequeueCount*)

setDestination

public void **setDestination** ([String](#) *destination*)

setEnqueueCount

public void **setEnqueueCount** (long *enqueueCount*)

setExpiredCount

public void **setExpiredCount** (long *expiredCount*)

setInflightCount

public void **setInflightCount** (long *inflightCount*)

setQueueSize

public void **setQueueSize** (long *queueSize*)

7.1.4 QueueMessage

public class **QueueMessage**

Represents a message from the JMS queue.

Constructors

QueueMessage

```
public QueueMessage (String messageId, Boolean redelivered, Date timestamp)
```

Methods

getMessageId

```
public String getMessageId ()
```

getRedelivered

```
public Boolean getRedelivered ()
```

getTimestamp

```
public String getTimestamp ()
```

7.1.5 StatusMessage

```
public class StatusMessage
```

Represents a message displayed in the ‘messages’ section of the Admin UI. Persisted by MDS module. Apart from the message and its `Level`, contains also information about the module that sent the message. The timeout field represents the `DateTime` of the message expiration.

Constructors

StatusMessage

```
public StatusMessage ()
```

StatusMessage

```
public StatusMessage (String text, String moduleName, Level level)
```

StatusMessage

```
public StatusMessage (String text, String moduleName, Level level, DateTime timeout)
```

Methods

getDate

```
public DateTime getDate ()
```

getLevel

public [Level](#) **getLevel** ()

getModuleName

public [String](#) **getModuleName** ()

getText

public [String](#) **getText** ()

getTimeout

public [DateTime](#) **getTimeout** ()

setDate

public void **setDate** ([DateTime](#) *date*)

setLevel

public void **setLevel** ([Level](#) *level*)

setModuleName

public void **setModuleName** ([String](#) *moduleName*)

setText

public void **setText** ([String](#) *text*)

setTimeout

public void **setTimeout** ([DateTime](#) *timeout*)

7.2 org.motechproject.admin.messages

7.2.1 ActionType

public enum **ActionType**

Represents an action which should be taken for a notification, in particular which message channel should be used to communicate the notification.

Enum Constants

EMAIL

public static final [ActionType](#) **EMAIL**

SMS

public static final [ActionType](#) **SMS**

7.2.2 Level

public enum **Level**

Represents the level of a `org.motechproject.admin.domain.StatusMessage`, which is reflected on the UI. Posting a `org.motechproject.admin.domain.StatusMessage` with a **CRITICAL** level will trigger notifications.

See also: `org.motechproject.admin.domain.StatusMessage`

Enum Constants

CRITICAL

public static final [Level](#) **CRITICAL**

DEBUG

public static final [Level](#) **DEBUG**

ERROR

public static final [Level](#) **ERROR**

INFO

public static final [Level](#) **INFO**

WARN

public static final [Level](#) **WARN**

7.3 org.motechproject.admin.service

7.3.1 NotificationRulesDataService

public interface **NotificationRulesDataService** extends [MotechDataService](#)<[NotificationRule](#)>
MDS data service for [NotificationRules](#).

7.3.2 StatusMessageService

public interface **StatusMessageService**

Methods

critical

void **critical** ([String](#) *text*, [String](#) *moduleName*)

critical

void **critical** ([String](#) *text*, [String](#) *moduleName*, [DateTime](#) *timeout*)

debug

void **debug** ([String](#) *text*, [String](#) *moduleName*)

debug

void **debug** ([String](#) *text*, [String](#) *moduleName*, [DateTime](#) *timeout*)

error

void **error** ([String](#) *text*, [String](#) *moduleName*)

error

void **error** ([String](#) *text*, [String](#) *moduleName*, [DateTime](#) *timeout*)

getActiveMessages

[List](#)<[StatusMessage](#)> **getActiveMessages** ()

getAllMessages

[List](#)<[StatusMessage](#)> **getAllMessages** ()

getNotificationRules

`List<NotificationRule> getNotificationRules ()`

info

`void info (String text, String moduleName)`

info

`void info (String text, String moduleName, DateTime timeout)`

postMessage

`void postMessage (StatusMessage message)`

postMessage

`void postMessage (String text, String moduleName, Level level)`

postMessage

`void postMessage (String text, String moduleName, Level level, DateTime timeout)`

removeMessage

`void removeMessage (StatusMessage message)`

removeNotificationRule

`void removeNotificationRule (String id)`

removeNotificationRule

`void removeNotificationRule (Long id)`

saveNotificationRules

`void saveNotificationRules (List<NotificationRule> notificationRules)`

saveRule

`void saveRule (NotificationRule notificationRule)`

warn

void **warn** (*String text*, *String moduleName*)

warn

void **warn** (*String text*, *String moduleName*, *DateTime timeout*)

7.3.3 StatusMessagesDataService

public interface **StatusMessagesDataService** extends *MotechDataService<StatusMessage>*
MDS data service for *StatusMessages*.

Methods

findByTimeout

List<StatusMessage> **findByTimeout** (*Range<DateTime> timeout*)

7.4 org.motechproject.bundle.extender

7.4.1 MotechExtenderConfigFactory

public class **MotechExtenderConfigFactory** implements *FactoryBean<Properties>*

Creates the extender configuration for Motech, currently only blueprint dependency wait time.
In order to change the blueprint extender dependency wait time used during platform runtime,
org.motechproject.blueprint.dependencies.waittime should be set with the wait time in
milliseconds. The default blueprint timeout is 5 minutes.

Fields

DEP_WAIT_TIME_ENV

public static final *String* **DEP_WAIT_TIME_ENV**

DEP_WAIT_TIME_KEY

public static final *String* **DEP_WAIT_TIME_KEY**

Methods

getObject

public *Properties* **getObject** ()

getObjectType

```
public Class<?> getObjectType ()
```

isSingleton

```
public boolean isSingleton ()
```

7.4.2 MotechOsgiApplicationContextCreator

```
public class MotechOsgiApplicationContextCreator implements OsgiApplicationContextCreator
```

Methods**createApplicationContext**

```
public DelegatedExecutionOsgiBundleApplicationContext createApplicationContext (BundleContext  
bundleCon-  
text)
```

setConfigurationScanner

```
public void setConfigurationScanner (ConfigurationScanner configurationScanner)
```

7.4.3 MotechOsgiConfigurableApplicationContext

```
public class MotechOsgiConfigurableApplicationContext extends OsgiBundleXmlApplicationContext implements Conf
```

Constructors**MotechOsgiConfigurableApplicationContext**

```
public MotechOsgiConfigurableApplicationContext (String[] configurationLocations)
```

Methods**getNamespace**

```
public String getNamespace ()
```

getServletConfig

```
public ServletConfig getServletConfig ()
```

getServletContext

public [ServletContext](#) **getServletContext** ()

setConfigLocation

public void **setConfigLocation** ([String](#) *configLocation*)

setNamespace

public void **setNamespace** ([String](#) *namespace*)

setServletConfig

public void **setServletConfig** ([ServletConfig](#) *servletConfig*)

setServletContext

public void **setServletContext** ([ServletContext](#) *servletContext*)

waitForContext

public void **waitForContext** (int *waitTimeInMillis*)

7.5 org.motechproject.commons.api

7.5.1 AbstractDataProvider

public abstract class **AbstractDataProvider** extends [MotechObject](#) implements [DataProvider](#)

Methods

getBody

protected [String](#) **getBody** ()

getClassForType

protected [Class](#)<?> **getClassForType** ([String](#) *type*)

getPackageRoot

public abstract [String](#) **getPackageRoot** ()

getSupportClasses

public abstract `List<Class<?>>` **getSupportClasses** ()

isAssignable

protected boolean **isAssignable** (`Class<?>` *check*, `List<Class<?>>` *classes*)

setBody

protected void **setBody** (`String` *body*)

setBody

protected void **setBody** (`Resource` *resource*)

supports

public boolean **supports** (`String` *type*)

toJSON

public `String` **toJSON** ()

7.5.2 ApplicationContextServiceReferenceUtils

public final class **ApplicationContextServiceReferenceUtils**

Fields

SERVICE_NAME

public static final `String` **SERVICE_NAME**

Methods

isValid

public static boolean **isValid** (`ServiceReference` *serviceReference*)

isNotValid

public static boolean **isNotValid** (`ServiceReference` *serviceReference*)

7.5.3 CastUtils

public final class **CastUtils**

Methods

cast

public static <T> **List**<T> **cast** (**Class**<T> *clazz*, **Enumeration** *enumeration*)

7.5.4 CsvConverter

public final class **CsvConverter**

The `CsvConverter` class, provides methods responsible for conversion to CSV-formatted strings.

Methods

convertToCSV

public static **String** **convertToCSV** (**List**<**List**<**String**>> *list*)

7.5.5 DataProvider

public interface **DataProvider**

Methods

getName

String **getName** ()

lookup

Object **lookup** (**String** *type*, **String** *lookupName*, **Map**<**String**, **String**> *lookupFields*)

supports

boolean **supports** (**String** *type*)

toJSON

String **toJSON** ()

7.5.6 IdentityProvider

public interface **IdentityProvider**

Methods

getIdentity

`String getIdentity()`

7.5.7 MotechEnumUtils

public final class **MotechEnumUtils**
Misc enum-related helper functions

Methods

toEnumSet

public static <T extends Enum> `Set<T> toEnumSet` (`Class<T> enumClass`, `Set<String> strings`)
Returns a set of enums given a set of strings and an enum class

Parameters

- **enumClass** – the enum class
- **strings** – a set of strings

Returns the enum set constructed from the given string set

toEnumSet

public static <T extends Enum> `Set<T> toEnumSet` (`Class<T> enumClass`, `String csv`)
Returns a set of enums given a comma separated string and an enum class

Parameters

- **enumClass** – the enum class
- **csv** – a comma separated string representing a set of enum values

Returns the enum set constructed from the given string

toString

public static `String toString` (`Set<? extends Enum> items`)
Returns a csv string given a set of enums

Parameters

- **items** – a set of enums

Returns the csv string constructed from the given enum set

toStringSet

public static `Set<String>` **toStringSet** (`Set<? extends Enum>` *items*)

Returns a set of strings given a set of enums

Parameters

- **items** – a set of enums

Returns the string set constructed from the given enum set

7.5.8 MotechException

public class **MotechException** extends `RuntimeException`

Constructors

MotechException

public **MotechException** (`String` *s*, `Throwable` *throwable*)

MotechException

public **MotechException** (`String` *message*)

7.5.9 MotechMapUtils

public final class **MotechMapUtils**

The `MotechMapUtils` class contains methods that allow modifications and operations on maps

Methods

asProperties

public static `Properties` **asProperties** (`Map<Object, Object>` *map*)

Converts `java.util.Map` into `java.util.Properties`

Parameters

- **map** – Map to convert

Returns Properties, created from given map

mergeMaps

public static `Map<Object, Object>` **mergeMaps** (`Map<Object, Object>` *overridingMap*, `Map<Object, Object>` *baseMap*)

Null-safe merge of two maps. If both parameters are null it returns empty map. If one of the maps is null, it returns the other one. If a key is present in two maps, the value in the merged map will be taken from the overriding map.

Parameters

- **overridingMap** – The map overriding values in the base map
- **baseMap** – The base map

Returns merged map

7.5.10 MotechObject

public class **MotechObject**

LayerSupertype for entire motech project. Used optionally through the motech project.

Methods

assertArgumentNotEmpty

protected void **assertArgumentNotEmpty** (*String* *objectName*, *String* *argument*)

assertArgumentNotNull

protected void **assertArgumentNotNull** (*String* *objectName*, *Object* *object*)

logError

protected void **logError** (*String* *templateMessage*, *Object...* *params*)

logError

protected void **logError** (*String* *message*)

logError

protected void **logError** (*String* *message*, *Exception* *e*)

logInfo

protected void **logInfo** (*String* *templateMessage*, *Object...* *params*)

logger

protected *Logger* **logger** ()

7.5.11 NanoStopWatch

public class **NanoStopWatch**

Constructors

NanoStopWatch

```
public NanoStopWatch ()
```

Methods

duration

```
public long duration ()
```

start

```
public NanoStopWatch start ()
```

7.5.12 Range

```
public class Range<T>
```

Constructors

Range

```
public Range (T min, T max)
```

Methods

equals

```
public boolean equals (Object o)
```

getMax

```
public T getMax ()
```

getMin

```
public T getMin ()
```

hashCode

```
public int hashCode ()
```

7.5.13 SystemIdentityProvider

public class **SystemIdentityProvider** implements [IdentityProvider](#)

Methods

getIdentity

public [String](#) **getIdentity**()

7.5.14 Tenant

public class **Tenant**

Constructors

Tenant

Tenant ([TenantIdentity](#) *identity*)

Methods

canHaveQueue

public boolean **canHaveQueue** ([String](#) *queueName*)

current

public static [Tenant](#) **current**()

getId

public [String](#) **getId**()

getSuffixedId

public [String](#) **getSuffixedId**()

7.5.15 TenantIdentity

public class **TenantIdentity**

Constructors

TenantIdentity

```
public TenantIdentity ()
```

TenantIdentity

```
public TenantIdentity (IdentityProvider identityProvider)
```

Methods

getId

```
public String getId ()
```

7.6 org.motechproject.commons.api.json

7.6.1 MotechJsonReader

```
public class MotechJsonReader
```

Constructors

MotechJsonReader

```
public MotechJsonReader ()
```

MotechJsonReader

```
public MotechJsonReader (FieldNamingStrategy fieldNamingStrategy)
```

Methods

readFromFile

```
public Object readFromFile (String classpathFile, Type ofType)
```

readFromStream

```
public Object readFromStream (InputStream stream, Type ofType)
```

readFromString

```
public Object readFromString (String text, Type ofType)
```

readFromString

```
public Object readFromString (String text, Type ofType, Map<Type, Object> typeAdapters)
```

7.7 org.motechproject.commons.api.model

7.7.1 ExtensibleDataObject

```
public abstract class ExtensibleDataObject<T>
```

Methods

addData

```
public T addData (String key, Object value)
```

addData

```
public T addData (Map<String, Object> data)
```

getData

```
public Map<String, Object> getData ()
```

7.7.2 MotechProperties

```
public class MotechProperties extends HashMap<String, String>
```

7.8 org.motechproject.commons.couchdb.dao

7.8.1 BaseDao

```
public interface BaseDao<T extends MotechBaseDataObject>
```

Methods

add

```
void add (T entity)
```

contains

```
boolean contains (String id)
```

get

T **get** (*String id*)

getAll

List<**T**> **getAll** ()

remove

void **remove** (*T entity*)

safeRemove

void **safeRemove** (*T entity*)

update

void **update** (*T entity*)

7.8.2 BusinessIdNotUniqueException

public class **BusinessIdNotUniqueException** extends *RuntimeException*

Constructors

BusinessIdNotUniqueException

public **BusinessIdNotUniqueException** (*String fieldName*, *String id*)

Methods

getFieldName

public *String* **getFieldName** ()

getId

public *String* **getId** ()

7.8.3 MotechBaseRepository

public abstract class **MotechBaseRepository**<**T** extends *MotechBaseDataObject*> extends *CouchDbRepositorySupport*<**T**>

Constructors

MotechBaseRepository

protected **MotechBaseRepository** (*Class*<T> *type*, CouchDbConnector *db*)

Methods

addOrReplace

protected void **addOrReplace** (T *entity*, *String* *businessFieldName*, *String* *businessId*)

bulkAddOrUpdate

public void **bulkAddOrUpdate** (*List*<T> *records*)

bulkDelete

public void **bulkDelete** (*List*<T> *records*)

getAll

public *List*<T> **getAll** ()

getAll

protected *List*<T> **getAll** (int *limit*)

queryViewWithKeyList

public *List*<T> **queryViewWithKeyList** (*String* *viewName*, *List*<*String*> *keys*)

removeAll

public void **removeAll** (*String* *fieldName*, *String* *value*)

removeAll

public void **removeAll** ()

safeRemove

public void **safeRemove** (T *entity*)

singleResult

protected T **singleResult** (*List<T> resultSet*)

7.9 org.motechproject.commons.couchdb.lucene.query

7.9.1 CouchDbLuceneQuery

public class **CouchDbLuceneQuery**

Fields

DATE_TIME_FORMAT

public static final *String* **DATE_TIME_FORMAT**

Methods

build

public *StringBuilder* **build** ()

with

public *CouchDbLuceneQuery* **with** (*String field*, *String value*)

withAny

public *CouchDbLuceneQuery* **withAny** (*String field*, *Set<String> values*)

withDate

public *CouchDbLuceneQuery* **withDate** (*String field*, *DateTime value*)

withDateRange

public *CouchDbLuceneQuery* **withDateRange** (*String field*, *Range<DateTime> value*)

withField

public *CouchDbLuceneQuery* **withField** (*String field*, *String type*, *String value*)

withInt

public CouchDbLuceneQuery **withInt** (*String field*, int *value*)

7.10 org.motechproject.commons.couchdb.model

7.10.1 MotechBaseDataObject

public abstract class **MotechBaseDataObject** extends CouchDbDocument

Constructors

MotechBaseDataObject

protected **MotechBaseDataObject** ()

MotechBaseDataObject

protected **MotechBaseDataObject** (*String type*)

Methods

getType

public *String* **getType** ()

setType

protected void **setType** (*String type*)

7.11 org.motechproject.commons.couchdb.query

7.11.1 QueryParam

public class **QueryParam**

Constructors

QueryParam

public **QueryParam** ()

QueryParam

public **QueryParam** (int *pageNumber*, int *recordsPerPage*, *String* *sortBy*, boolean *reverse*)

Methods

getPageNumber

public int **getPageNumber** ()

getRecordsPerPage

public int **getRecordsPerPage** ()

getSortBy

public *String* **getSortBy** ()

isReverse

public boolean **isReverse** ()

setPageNumber

public void **setPageNumber** (int *pageNumber*)

setRecordsPerPage

public void **setRecordsPerPage** (int *recordsPerPage*)

setReverse

public void **setReverse** (boolean *reverse*)

setSortBy

public void **setSortBy** (*String* *sortBy*)

7.12 org.motechproject.commons.couchdb.service

7.12.1 CouchDbManager

public interface **CouchDbManager**

Methods

getConnector

CouchDbConnector **getConnector** (*String dbName*)

7.12.2 DbConnectionException

public class **DbConnectionException** extends *RuntimeException*

Constructors

DbConnectionException

public **DbConnectionException** (*String message*)

DbConnectionException

public **DbConnectionException** (*String message, Throwable cause*)

7.13 org.motechproject.commons.date

7.13.1 ParseException

public class **ParseException** extends *RuntimeException*

Constructors

ParseException

public **ParseException** (*String s*)

7.14 org.motechproject.commons.date.model

7.14.1 DayOfWeek

public enum **DayOfWeek**

Enum Constants

Friday

public static final *DayOfWeek* **Friday**

Monday

public static final [DayOfWeek](#) **Monday**

Saturday

public static final [DayOfWeek](#) **Saturday**

Sunday

public static final [DayOfWeek](#) **Sunday**

Thursday

public static final [DayOfWeek](#) **Thursday**

Tuesday

public static final [DayOfWeek](#) **Tuesday**

Wednesday

public static final [DayOfWeek](#) **Wednesday**

7.14.2 Time

public class **Time** implements [Comparable<Time>](#), [Serializable](#)

Constructors

Time

public **Time** ()

Time

public **Time** (int *hour*, int *minute*)

Time

public **Time** ([Integer](#) *hour*, [Integer](#) *minute*)

Time

public **Time** ([LocalTime](#) *localTime*)

Time

public **Time** (*String timeStr*)

Methods

compareTo

public int **compareTo** (*Time otherTime*)

equals

public boolean **equals** (*Object obj*)

ge

public boolean **ge** (*Time toCompare*)

getDateTime

public *DateTime* **getDateTime** (*DateTime dateTime*)

getHour

public *Integer* **getHour** ()

getMinute

public *Integer* **getMinute** ()

hashCode

public int **hashCode** ()

isBefore

public boolean **isBefore** (*Time other*)

le

public boolean **le** (*Time toCompare*)

parseTime

public static *Time* **parseTime** (*String time*, *String separator*)

setHour

public void **setHour** (*Integer hour*)

setMinute

public void **setMinute** (*Integer minute*)

timeStr

public *String* **timeStr** ()

toString

public *String* **toString** ()

valueOf

public static *Time* **valueOf** (*String str*)

7.15 org.motechproject.commons.date.util

7.15.1 DateTimeSourceUtil

public final class **DateTimeSourceUtil**

Methods

now

public static *DateTime* **now** ()

setSourceInstance

public static void **setSourceInstance** (*DateTimeSource sourceInstance*)

timeZone

public static *DateTimeZone* **timeZone** ()

today

public static *LocalDate* **today** ()

7.15.2 DateUtil

public final class **DateUtil**

Methods

daysPast

public static int **daysPast** (*LocalDate* *localDate*, *DayOfWeek* *dayOfWeek*)

daysStarting

public static List<*DayOfWeek*> **daysStarting** (*DayOfWeek* *day*, int *numberOfDays*)

daysToCalendarWeekEnd

public static int **daysToCalendarWeekEnd** (*LocalDate* *date*, int *calendarWeekStartDay*)

endOfDay

public static *DateTime* **endOfDay** (*Date* *dateTime*)

getDifferenceOfDatesInYears

public static int **getDifferenceOfDatesInYears** (*Date* *startDate*)

greaterThanOrEqualTo

public static List<*DateTime*> **greaterThanOrEqualTo** (*DateTime* *date*, List<*DateTime*> *dates*)

inRange

public static boolean **inRange** (*DateTime* *reference*, *DateTime* *start*, *DateTime* *end*)

isOnOrAfter

public static boolean **isOnOrAfter** (*DateTime* *firstDate*, *DateTime* *secondDate*)

isOnOrBefore

public static boolean **isOnOrBefore** (*DateTime* *firstDate*, *DateTime* *secondDate*)

lessThan

public static List<*DateTime*> **lessThan** (*DateTime* *date*, List<*DateTime*> *dates*)

newDate

public static `LocalDate` **newDate** (int *year*, int *month*, int *day*)

newDate

public static `LocalDate` **newDate** (`Date` *date*)

newDate

public static `LocalDate` **newDate** (`DateTime` *dateTime*)

newDateTime

public static `DateTime` **newDateTime** (`LocalDate` *localDate*, int *hour*, int *minute*, int *second*)

newDateTime

public static `DateTime` **newDateTime** (`Date` *date*)

newDateTime

public static `DateTime` **newDateTime** (`LocalDate` *date*)

newDateTime

public static `DateTime` **newDateTime** (`LocalDate` *localDate*, `Time` *time*)

newDateTime

public static `DateTime` **newDateTime** (int *year*, int *month*, int *day*, `Time` *time*)

newDateTime

public static `DateTime` **newDateTime** (int *year*, int *month*, int *day*)

newDateTime

public static `DateTime` **newDateTime** (int *year*, int *month*, int *day*, int *hour*, int *minute*, int *second*)

nextApplicableWeekDay

public static `DateTime` **nextApplicableWeekDay** (`DateTime` *fromDate*, `List<DayOfWeek>` *applicable-Days*)

nextApplicableWeekDayIncludingFromDate

```
public static DateTime nextApplicableWeekDayIncludingFromDate (DateTime fromDate,
                                                             List<DayOfWeek> applicableDays)
```

now

```
public static DateTime now ()
```

nowUTC

```
public static DateTime nowUTC ()
```

setTimeZone

```
public static DateTime setTimeZone (DateTime dateTime)
```

setTimeZoneUTC

```
public static DateTime setTimeZoneUTC (DateTime dateTime)
```

time

```
public static Time time (DateTime dateTime)
```

today

```
public static LocalDate today ()
```

tomorrow

```
public static LocalDate tomorrow ()
```

7.15.3 JodaFormatter

```
public class JodaFormatter
```

Constructors

JodaFormatter

```
public JodaFormatter ()
```

Methods

formatDateTime

public [String](#) **formatDateTime** ([DateTime](#) *dateTime*)

formatPeriod

public [String](#) **formatPeriod** ([Period](#) *period*)
Format joda period as text, eg: “1 year”

Parameters

- **period** – time interval

parse

public [Period](#) **parse** ([String](#) *intervalString*, [Locale](#) *locale*)
Parse time interval in different units, eg: “1 year”

Parameters

- **intervalString** – time interval format number: integer unit : year, month, week, day, hour, minute, second (can use plural forms also) currently compound units like 1 year and 2 months are not supported

parseDateTime

public [DateTime](#) **parseDateTime** ([String](#) *isoDateTime*)

parsePeriod

public [Period](#) **parsePeriod** ([String](#) *intervalString*)

7.15.4 StringUtil

public final class **StringUtil**

Methods

isNullOrEmpty

public static boolean **isNullOrEmpty** ([String](#) *str*)

7.16 org.motechproject.commons.date.util.datetime

7.16.1 DateTimeSource

public interface **DateTimeSource**

Methods

now

`DateTime now ()`

timeZone

`DateTimeZone timeZone ()`

today

`LocalDate today ()`

7.16.2 DefaultDateTimeSource

public class **DefaultDateTimeSource** implements `DateTimeSource`

Constructors

DefaultDateTimeSource

public **DefaultDateTimeSource** ()

Methods

now

public `DateTime now ()`

timeZone

public `DateTimeZone timeZone ()`

today

public `LocalDate today ()`

7.17 org.motechproject.commons.date.valueobjects

7.17.1 WallTime

public class **WallTime** implements [Serializable](#)

Constructors

WallTime

public **WallTime** ([String](#) *userReadableForm*)

Methods

asPeriod

public [Period](#) **asPeriod** ()

equals

public boolean **equals** ([Object](#) *o*)

getHours

public int **getHours** ()

getMinutes

public int **getMinutes** ()

hashCode

public int **hashCode** ()

inDays

public int **inDays** ()

inMillis

public long **inMillis** ()

inMinutes

public int **inMinutes** ()

isLessThanADay

```
public boolean isLessThanADay ()
```

7.17.2 WallTimeUnit

```
public enum WallTimeUnit
```

Enum Constants

Day

```
public static final WallTimeUnit Day
```

Hour

```
public static final WallTimeUnit Hour
```

Minute

```
public static final WallTimeUnit Minute
```

Week

```
public static final WallTimeUnit Week
```

7.18 org.motechproject.commons.sql.service

7.18.1 SqlDBManager

```
public interface SqlDBManager
```

Classes inheriting this interface are responsible for getting sql properties from the bootstrap configuration.

Methods

getSqlProperties

```
Properties getSqlProperties (Properties propertiesToUpdate)
```

updateSqlProperties

```
void updateSqlProperties ()
```

7.19 org.motechproject.config.core

7.19.1 MotechConfigurationException

public class **MotechConfigurationException** extends [RuntimeException](#)

The object of this class is thrown when there is a problem with reading the configuration from the predefined sources.

Constructors

MotechConfigurationException

public **MotechConfigurationException** ([String](#) message, [Exception](#) exception)

Parameters

- **message** – A descriptive message explaining the nature of the problem resulted in exception
- **exception** – Actual exception (if any) that this exception resulted from

MotechConfigurationException

public **MotechConfigurationException** ([String](#) message)

7.20 org.motechproject.config.core.constants

7.20.1 ConfigurationConstants

public final class **ConfigurationConstants**

Provides all the configuration constants.

Fields

AMQ_BROKER_URL

public static final [String](#) **AMQ_BROKER_URL**

AMQ_CONCURRENT_CONSUMERS

public static final [String](#) **AMQ_CONCURRENT_CONSUMERS**

AMQ_MAX_CONCURRENT_CONSUMERS

public static final [String](#) **AMQ_MAX_CONCURRENT_CONSUMERS**

AMQ_MAX_REDELIVERIES

public static final `String` **AMQ_MAX_REDELIVERIES**

AMQ_QUEUE_EVENTS

public static final `String` **AMQ_QUEUE_EVENTS**

AMQ_QUEUE_SCHEDULER

public static final `String` **AMQ_QUEUE_SCHEDULER**

AMQ_REDELIVERY_DELAY_IN_MILLIS

public static final `String` **AMQ_REDELIVERY_DELAY_IN_MILLIS**

BUNDLE_ID

public static final `String` **BUNDLE_ID**

BUNDLE_SECTION

public static final `String` **BUNDLE_SECTION**

BUNDLE_SETTINGS_CHANGED_EVENT_SUBJECT

public static final `String` **BUNDLE_SETTINGS_CHANGED_EVENT_SUBJECT**

BUNDLE_SYMBOLIC_NAME

public static final `String` **BUNDLE_SYMBOLIC_NAME**

CONFIG_MODULE_DIR_PREFIX

public static final `String` **CONFIG_MODULE_DIR_PREFIX**

EVENT_RELAY_CLASS_NAME

public static final `String` **EVENT_RELAY_CLASS_NAME**

FILE_CHANGED_EVENT_SUBJECT

public static final `String` **FILE_CHANGED_EVENT_SUBJECT**

FILE_CREATED_EVENT_SUBJECT

public static final [String](#) **FILE_CREATED_EVENT_SUBJECT**

FILE_DELETED_EVENT_SUBJECT

public static final [String](#) **FILE_DELETED_EVENT_SUBJECT**

FILE_PATH

public static final [String](#) **FILE_PATH**

LANGUAGE

public static final [String](#) **LANGUAGE**

LOGINMODE

public static final [String](#) **LOGINMODE**

MOTECH_EVENT_CLASS_NAME

public static final [String](#) **MOTECH_EVENT_CLASS_NAME**

PLATFORM_SETTINGS_CHANGED_EVENT_SUBJECT

public static final [String](#) **PLATFORM_SETTINGS_CHANGED_EVENT_SUBJECT**

PROVIDER_NAME

public static final [String](#) **PROVIDER_NAME**

PROVIDER_URL

public static final [String](#) **PROVIDER_URL**

SERVER_URL

public static final [String](#) **SERVER_URL**

SETTINGS

public static final [String](#) **SETTINGS**

SETTINGS_FILE_NAME

```
public static final String SETTINGS_FILE_NAME
```

STATUS_MSG_TIMEOUT

```
public static final String STATUS_MSG_TIMEOUT
```

SUPPORTED_FILE_EXTNS

```
public static final String[] SUPPORTED_FILE_EXTNS
```

UPLOAD_SIZE

```
public static final String UPLOAD_SIZE
```

7.21 org.motechproject.config.core.domain

7.21.1 AbstractDBConfig

```
public abstract class AbstractDBConfig
```

This abstract class encapsulates the database configuration, composed of as db url, username and password.

Constructors

AbstractDBConfig

```
public AbstractDBConfig (String url, String driver, String username, String password)
```

Parameters

- **url** –
- **driver** –
- **username** –
- **password** –

Methods

equals

```
public boolean equals (Object o)
```

getDriver

```
public String getDriver ()
```

getPassword

```
public String getPassword ()
```

getUrl

```
public String getUrl ()
```

getUsername

```
public String getUsername ()
```

hashCode

```
public int hashCode ()
```

toString

```
public String toString ()
```

7.21.2 BootstrapConfig

```
public class BootstrapConfig
```

Represents the bootstrap configuration object. It is composed of:

- 1.DBConfig - represents the database related bootstrap object.
- 2.Tenant ID - represents the identifier of the tenant.
- 3.Configuration source - represents the source of configuration (FILE / UI).

Fields

CONFIG_SOURCE

```
public static final String CONFIG_SOURCE
```

COUCHDB_PASSWORD

```
public static final String COUCHDB_PASSWORD
```

COUCHDB_URL

```
public static final String COUCHDB_URL
```

COUCHDB_USERNAME

```
public static final String COUCHDB_USERNAME
```

DEFAULT_TENANT_ID

```
public static final String DEFAULT_TENANT_ID
```

SQL_DRIVER

```
public static final String SQL_DRIVER
```

SQL_PASSWORD

```
public static final String SQL_PASSWORD
```

SQL_URL

```
public static final String SQL_URL
```

SQL_USER

```
public static final String SQL_USER
```

TENANT_ID

```
public static final String TENANT_ID
```

Constructors**BootstrapConfig**

```
public BootstrapConfig(DBConfig couchDbConfig, SQLDBConfig sqlConfig, String tenantId, Config-  
Source configSource)
```

Parameters

- **couchDbConfig** –
- **sqlConfig** –
- **tenantId** –
- **configSource** –

Throws

- **org.motechproject.config.core.MotechConfigurationException** – if dbConfig is null.

Methods

equals

public boolean **equals** (*Object o*)

getConfigSource

public *ConfigSource* **getConfigSource** ()

getCouchDbConfig

public *DBConfig* **getCouchDbConfig** ()

getSqlConfig

public *SQLDBConfig* **getSqlConfig** ()

getTenantId

public *String* **getTenantId** ()

hashCode

public int **hashCode** ()

toString

public *String* **toString** ()

7.21.3 ConfigLocation

public class **ConfigLocation**

Defines a MOTECH configuration location. If the given location starts with a leading file separator character, the location is treated as a file system directory. Otherwise, it is treated as a classpath location.

Constructors

ConfigLocation

public **ConfigLocation** (*String configLocation*)

Methods

equals

public boolean **equals** (*Object o*)

getExistingConfigFiles

public *List<File>* **getExistingConfigFiles** ()

getFile

public *File* **getFile** (*String fileName*, *FileAccessType accessType*)

This method Returns the *java.io.File* object for the given file name relative to the config location. It also checks for the requested file accessibility. If the requested access type check is *ConfigLocation.FileAccessType.READABLE*, the file's existence and readability will be checked. Similarly, if the requested access type check is *ConfigLocation.FileAccessType.WRITABLE*, then the write accessibility to the file will be checked. If the file does not exists, write accessibility of its ancestors will be checked.

Parameters

- **fileName** – Name of the file to be added to the config location.
- **accessType** – One of *ConfigLocation.FileAccessType.READABLE* or *ConfigLocation.FileAccessType.WRITABLE*.

Throws

- **MotechConfigurationException** – if the file is not readable or writable depending on the given access type.

Returns File relative to the config location.

getLocation

public *String* **getLocation** ()

getUrlResource

UrlResource **getUrlResource** ()

hasPlatformConfigurationFile

public boolean **hasPlatformConfigurationFile** ()

hashCode

public int **hashCode** ()

toResource

public [Resource](#) **toResource** ()
Resource corresponding to the config location.
Returns resource

toString

public [String](#) **toString** ()

7.21.4 ConfigLocation.FileAccessType

public static enum **FileAccessType**
Defines the access check required.

Enum Constants

READABLE

public static final [ConfigLocation.FileAccessType](#) **READABLE**

WRITABLE

public static final [ConfigLocation.FileAccessType](#) **WRITABLE**

7.21.5 ConfigSource

public final class **ConfigSource**
Represents the source from which MOTECH configuration should be read.

Fields

FILE

public static final [ConfigSource](#) **FILE**

UI

public static final [ConfigSource](#) **UI**

Methods

getName

public [String](#) **getName** ()

isFile

```
public boolean isFile ()
```

isValid

```
public static boolean isValid (String name)
```

toString

```
public String toString ()
```

valueOf

```
public static ConfigSource valueOf (String name)
```

7.21.6 DBConfig

```
public class DBConfig extends AbstractDBConfig
```

DBConfig encapsulates the database configuration, composed of as db url, username and password.

Constructors

DBConfig

```
public DBConfig (String url, String username, String password)
```

Parameters

- **url** –
- **username** –
- **password** –

Throws

- **org.motechproject.config.core.MotechConfigurationException** – if given url is invalid.

7.21.7 SQLDBConfig

```
public class SQLDBConfig extends AbstractDBConfig
```

This class encapsulates the SQL database configuration, composed of as db url, username and password.

Constructors

SQLDBConfig

```
public SQLDBConfig (String url, String driver, String username, String password)
```

Parameters

- **url** –
- **driver** –
- **username** –
- **password** –

Throws

- **org.motechproject.config.core.MotechConfigurationException** – if given url is invalid.

7.22 org.motechproject.config.core.filestore

7.22.1 ConfigLocationFileStore

public class **ConfigLocationFileStore**

Used to read default platform config location(s) from `config-location.properties` and also to save in the file in the default location.

`config-location.properties` file will be loaded according to the behaviour of `org.apache.commons.configuration.PropertiesConfiguration` as specified [here](#).

Fields

CONFIG_LOCATION_PROPERTY_KEY

public static final [String](#) **CONFIG_LOCATION_PROPERTY_KEY**

Constructors

ConfigLocationFileStore

public **ConfigLocationFileStore** ([PropertiesConfiguration](#) *propertiesConfiguration*)

Methods

add

public void **add** ([String](#) *location*)

getAll

public [Iterable](#)<[ConfigLocation](#)> **getAll** ()

7.22.2 PropertiesReader

public final class **PropertiesReader**

A utility class for loading properties from a file.

Methods

getProperties

public static [Properties](#) **getProperties** ([File](#) *file*)

7.23 org.motechproject.config.core.filters

7.23.1 ConfigFileFilter

public class **ConfigFileFilter** extends [FileFileFilter](#)
FileFilter implementation to filter configuration files.

Fields

PLATFORM_CORE_CONFIG_FILTER

public static final [FileFileFilter](#) **PLATFORM_CORE_CONFIG_FILTER**

Methods

accept

public boolean **accept** ([File](#) *file*)

isFileSupported

public static boolean **isFileSupported** ([File](#) *file*)

isPlatformCoreConfigFile

public static boolean **isPlatformCoreConfigFile** ([File](#) *file*)

7.24 org.motechproject.config.core.service

7.24.1 CoreConfigurationService

public interface **CoreConfigurationService**
Loads and saves the core configuration required to start the Motech instance.

Fields

CORE_SETTINGS_CACHE_NAME

[String](#) **CORE_SETTINGS_CACHE_NAME**

Methods

addConfigLocation

void **addConfigLocation** (*String location*)

Adds the new config location to the list of existing config locations where configurations are loaded from in the file system.

Parameters

- **location** – config location to add.

Throws

- **FileSystemException** –

evictMotechCoreSettingsCache

void **evictMotechCoreSettingsCache** ()

getConfigLocation

ConfigLocation **getConfigLocation** ()

Returns the config location where all the config files are present.

Returns *configLocation*.

loadBootstrapConfig

BootstrapConfig **loadBootstrapConfig** ()

Loads the bootstrap configuration.

Returns bootstrap configuration.

saveBootstrapConfig

void **saveBootstrapConfig** (*BootstrapConfig bootstrapConfig*)

Saves the bootstrap configuration

Parameters

- **bootstrapConfig** – Bootstrap config

7.25 org.motechproject.config.domain

7.25.1 ModulePropertiesRecord

public class **ModulePropertiesRecord**

The `ModulePropertiesRecord` class represents a database record of a certain module properties.

Fields

PROPERTIES_FILE_EXTENSION

```
public static final String PROPERTIES_FILE_EXTENSION
```

Constructors

ModulePropertiesRecord

```
public ModulePropertiesRecord ()
```

ModulePropertiesRecord

```
public ModulePropertiesRecord (Map<String, String> properties, String module, String version, String bundle, String filename, boolean raw)
```

ModulePropertiesRecord

```
public ModulePropertiesRecord (Properties props, String module, String version, String bundle, String filename, boolean raw)
```

Methods

buildFrom

```
public static ModulePropertiesRecord buildFrom (File file)
```

equals

```
public boolean equals (Object obj)
```

getBundle

```
public String getBundle ()
```

getFilename

```
public String getFilename ()
```

getModule

```
public String getModule ()
```

getProperties

```
public Map<String, String> getProperties ()
```

getVersion

```
public String getVersion ()
```

hashCode

```
public int hashCode ()
```

isRaw

```
public boolean isRaw ()
```

sameAs

```
public boolean sameAs (Object dataObject)
```

setBundle

```
public void setBundle (String bundle)
```

setFilename

```
public void setFilename (String filename)
```

setModule

```
public void setModule (String module)
```

setProperties

```
public void setProperties (Map<String, String> properties)
```

setRaw

```
public void setRaw (boolean raw)
```

setVersion

```
public void setVersion (String version)
```

toString

```
public String toString ()
```

7.26 org.motechproject.config.monitor

7.26.1 ConfigFileMonitor

public class **ConfigFileMonitor** implements **FileListener**

The **ConfigFileMonitor** is used to monitor changes in config files and send appropriate events.

Methods

fileChanged

```
public void fileChanged (FileChangeEvent fileChangeEvent)
```

fileCreated

```
public void fileCreated (FileChangeEvent fileChangeEvent)
```

fileDeleted

```
public void fileDeleted (FileChangeEvent fileChangeEvent)
```

init

```
public void init ()
```

setFileMonitor

```
public void setFileMonitor (DefaultFileMonitor fileMonitor)
```

stop

```
public void stop ()
```

updateFileMonitor

```
public void updateFileMonitor ()
```

7.27 org.motechproject.config.service

7.27.1 ConfigurationService

public interface **ConfigurationService**

Central configuration service that monitors and manages configurations.

Fields

SETTINGS_CACHE_NAME

String **SETTINGS_CACHE_NAME**

Methods

addOrUpdate

void **addOrUpdate** (File *file*)

Saves both property and raw configurations in FILE mode only. Files are classified as either raw config or properties based on the extension of the file.

Parameters

- **file** – File to read configuration from.

addOrUpdateModuleRecord

void **addOrUpdateModuleRecord** (ModulePropertiesRecord *record*)

addOrUpdateModuleRecords

void **addOrUpdateModuleRecords** (List<ModulePropertiesRecord> *records*)

addOrUpdateProperties

void **addOrUpdateProperties** (String *module*, String *version*, String *bundle*, String *filename*, Properties *newProperties*, Properties *defaultProperties*)

Depending on the config source, it will either store properties in the DB or file. Only properties that are different from the default ones are stored. If the properties database record or file doesn't exist yet for the given module, it will be created.

Parameters

- **module** – The module we wish to update properties for
- **version** – Version of updated bundle
- **bundle** – Symbolic name of updated bundle
- **filename** – Resource filename
- **newProperties** – New properties to store

- **defaultProperties** – Default properties of the module

Throws

- **IOException** – if module properties cannot be retrieved from file

createZipWithConfigFiles

`FileInputStream createZipWithConfigFiles (String propertyFile, String fileName)`

Uses current configuration and default one to find changed properties and then connects them with annotations. Moreover creates file with non default configurations and packs it into the zip file.

Parameters

- **propertyFile** – name of exported file

Throws

- **IOException** –

Returns `FileInputStream` that contains zip file

delete

`void delete (String module)`

Deletes the db record corresponding to the module.

deleteByBundle

`void deleteByBundle (String module)`

Deletes the db record corresponding to the module with given bundle symbolic name.

evictMotechSettingsCache

`void evictMotechSettingsCache ()`

getAllModuleProperties

`Map<String, Properties> getAllModuleProperties (String module, Map<String, Properties> defaultProperties)`

Retrieves all the module properties and returns them as Map, where key is the filename.

Parameters

- **module** – The module we wish to retrieve properties for
- **defaultProperties** – Default properties of the module

Throws

- **IOException** – if any of the module properties file cannot be read

Returns Properties mapped by filename

getConfigSource

`ConfigSource` **getConfigSource** ()

This method allows to check whether MOTECH is currently running in the FILE or UI mode

Returns Current Config Source

getModuleProperties

`Properties` **getModuleProperties** (`String module`, `String filename`, `Properties defaultProperties`)

Retrieves merged properties, given default set. Depending on the ConfigSource, it will either merge default properties with the properties from DB or get properties from file.

Parameters

- **module** – The module we wish to retrieve properties for
- **filename** – Resource filename
- **defaultProperties** – Default properties of the module

Throws

- **IOException** – if module properties cannot be read from file

Returns Merged properties of the certain module

getPlatformSettings

`MotechSettings` **getPlatformSettings** ()

getRawConfig

`InputStream` **getRawConfig** (`String module`, `String filename`, `Resource resource`)

Allows to retrieve raw JSON data either from the database or file, depending on the specified ConfigSource mode.

Parameters

- **module** – Module we wish to retrieve raw data for
- **filename** – Resource filename
- **resource** – Resource file containing default rawConfig, in case no other has been found

Throws

- **IOException** –

Returns Raw JSON data as InputStream

listRawConfigNames

`List<String>` **listRawConfigNames** (`String module`)

Depending on the selected ConfigSource mode, this method looks for all registered raw data properties within the specified module.

Parameters

- **module** – Module we wish to perform look for

Returns List of filenames that register raw config for specified module

loadBootstrapConfig

BootstrapConfig loadBootstrapConfig()

Loads bootstrap config that is used to start up the Motech server.

The bootstrap configuration is loaded in the following order:

1. Load the configuration from `bootstrap.properties` from the config directory specified by the environment variable `MOTech_CONFIG_DIR`. `bootstrap.properties` contains the following properties:

```
couchDb.url (Mandatory)
couchDb.username (If required)
couchDb.password (If required)
sql.url (Mandatory)
sql.username (If required)
sql.password (If required)
tenant.id (Optional. Defaults to 'DEFAULT')
config.source (Optional. Defaults to 'UI')
```

An example `bootstrap.properties` is given below:

```
couchDb.url=http://localhost:5984
couchDb.username=motech
couchDb.password=motech
sql.url=jdbc:mysql://localhost:3306/
sql.username=motech
sql.password=motech
tenant.id=MotherChildCare
config.source=FILE
```

2. If `MOTech_CONFIG_DIR` environment variable is **not** set, load the specific configuration values from the following environment variables:

```
MOTech_COUCHDB_URL (Mandatory)
MOTech_COUCHDB_USERNAME (If required)
MOTech_COUCHDB_PASSWORD (If required)
MOTech_SQL_URL (Mandatory)
MOTech_SQL_USERNAME (If required)
MOTech_SQL_PASSWORD (If required)
MOTech_TENANT_ID (Optional. Defaults to 'DEFAULT')
MOTech_CONFIG_SOURCE (Optional. Defaults to 'UI')
```

3. If `MOTech_DB_URL` environment is not set, load the configuration from `bootstrap.properties` from the default MOTech config directory specified in the file `config-locations.properties`.

Throws

- [org.motechproject.config.core.MotechConfigurationException](#) – if bootstrap configuration cannot be loaded.

Returns Bootstrap configuration

loadConfig

`SettingsRecord loadConfig()`

loadDefaultConfig

`SettingsRecord loadDefaultConfig()`

processExistingConfigs

`void processExistingConfigs(List<File> files)`

Adds, updates, or deletes configurations in FILE mode only. Files are classified as either raw config or properties based on the extension of the file.s

Parameters

- **files** – Files to read configuration from.

rawConfigExists

`boolean rawConfigExists(String module, String filename)`

Allows to check if raw data has been registered for specified module

Parameters

- **module** – Module symbolic name
- **filename** – Resource filename

Returns True if raw data exists for given parameters, false otherwise

registersProperties

`boolean registersProperties(String module, String filename)`

Checks if given module registers certain property file

Parameters

- **module** – Module we wish to perform check for
- **filename** – Resource filename

Returns True if properties exist, false otherwise

removeModuleRecords

`void removeModuleRecords(List<ModulePropertiesRecord> records)`

removeProperties

`void removeProperties(String module, String filename)`

Removes properties for given module from database or file.

Parameters

- **module** – The module we wish to remove properties for
- **filename** – Resource filename

requiresConfigurationFiles

boolean **requiresConfigurationFiles** ()

retrieveRegisteredBundleNames

List<String> **retrieveRegisteredBundleNames** ()

Depending on the selected ConfigSource mode, this method looks for registered bundle properties and returns a list of files it has found

Returns List of files with registered properties

save

void **save** (*BootstrapConfig bootstrapConfig*)

Saves the given *BootstrapConfig* in the *bootstrap.properties* file located in default MOTECH config location. The default motech config location is specified in the file *config-locations.properties*.

Parameters

- **bootstrapConfig** – Bootstrap configuration.

Throws

- **org.motechproject.config.core.MotechConfigurationException** – if bootstrap configuration cannot be saved.

savePlatformSettings

void **savePlatformSettings** (*Properties settings*)

savePlatformSettings

void **savePlatformSettings** (*MotechSettings settings*)

saveRawConfig

void **saveRawConfig** (*String module*, *String version*, *String bundle*, *String filename*, *InputStream rawData*)

Allows persisting of raw json properties either in the database or file, depending on the selected ConfigSource mode.

Parameters

- **module** – Module we wish to save properties for
- **filename** – Resource filename
- **rawData** – Raw JSON data to persist

Throws

- **IOException** –

setPlatformSetting

void **setPlatformSetting** (*String* key, *String* value)

updateConfigLocation

void **updateConfigLocation** (*String* newConfigLocation)
Adds a new config location and restarts the monitor.

Parameters

- **newConfigLocation** – New config location

updatePropertiesAfterReinstallation

void **updatePropertiesAfterReinstallation** (*String* module, *String* version, *String* bundle, *String* filename, *Properties* defaultProperties, *Properties* newProperties)

Works similar to `addOrUpdateProperties` but instead of just adding / updating properties checks database for any deprecated properties and removes to ensure that only current ones are available

Parameters

- **module** – The module we wish to update properties for
- **version** – Version of updated bundle
- **bundle** – Symbolic name of updated bundle
- **filename** – Resource filename
- **newProperties** – New properties to store
- **defaultProperties** – Default properties of the module

Throws

- **IOException** – if module properties cannot be retrieved from file

7.27.2 ModulePropertiesService

public interface **ModulePropertiesService** extends [MotechDataService<ModulePropertiesRecord>](#)

Methods**findByBundle**

[List<ModulePropertiesRecord>](#) **findByBundle** (*String* bundle)

findByModule

List<ModulePropertiesRecord> **findByModule** (String *module*)

findByModuleAndFileName

List<ModulePropertiesRecord> **findByModuleAndFileName** (String *module*, String *filename*)

7.28 org.motechproject.email.builder

7.28.1 EmailRecordSearchCriteria

public class **EmailRecordSearchCriteria**

The EmailRecordSearchCriteria class represents search criteria that may be used for searching `org.motechproject.email.domain.EmailRecord` entities in Motech Data Services. A consumer of this class may create search criteria to query on multiple fields by calling several of the with* methods. To perform the search, use `org.motechproject.email.service.EmailAuditService.findEmailRecords (EmailRecordSearchCr`

Methods

getDeliveryStatuses

public Set<DeliveryStatus> **getDeliveryStatuses** ()

Gets the delivery statuses criterion.

Returns the delivery statuses criterion for this search criteria

getDeliveryTimeRange

public Range<DateTime> **getDeliveryTimeRange** ()

Gets the delivery time range criterion.

Returns the deliveryTimeRange criterion for this search criteria

getFromAddress

public String **getFromAddress** ()

Gets the from address criterion.

Returns the fromAddress criterion for this search criteria

getMessage

public String **getMessage** ()

Gets the message body criterion.

Returns the message criterion for this search criteria

getQueryParams

public [QueryParams](#) **getQueryParams** ()

Gets the query paramaters that are used for controlling order and size of the query results for this search criteria.

Returns the query params for this search criteria

getSubject

public [String](#) **getSubject** ()

Gets the subject criterion.

Returns the subject criterion for this search criteria

getToAddress

public [String](#) **getToAddress** ()

Gets the to address criterion.

Returns the toAddress criterion for this search criteria

withDeliveryStatuses

public [EmailRecordSearchCriteria](#) **withDeliveryStatuses** ([Set](#)<[DeliveryStatus](#)> *deliveryStatuses*)

Sets the delivery statuses criterion to the set specified

Parameters

- **deliveryStatuses** – the delivery statuses on which to search

Returns this [EmailRecordSearchCriteria](#) with its [deliveryStatuses](#) criterion set to the provided statuses

withDeliveryStatuses

public [EmailRecordSearchCriteria](#) **withDeliveryStatuses** ([DeliveryStatus](#)... *deliveryStatuses*)

Sets the delivery statuses criterion to the set specified

Parameters

- **deliveryStatuses** – the delivery statuses on which to search

Returns this [EmailRecordSearchCriteria](#) with its [deliveryStatuses](#) criterion set to the provided statuses

withFromAddress

public [EmailRecordSearchCriteria](#) **withFromAddress** ([String](#) *fromAddress*)

Sets the fromAddress criterion to the address specified

Parameters

- **fromAddress** – the sender email address on which to search

Returns this `EmailRecordSearchCriteria` with its `fromAddress` criterion set to the provided address

`withMessage`

public `EmailRecordSearchCriteria` **withMessage** (`String` *message*)
Sets the message criterion to the message specified

Parameters

- **message** – the email message body on which to search

Returns this `EmailRecordSearchCriteria` with its message criterion set to the provided message

`withMessageTime`

public `EmailRecordSearchCriteria` **withMessageTime** (`DateTime` *deliveryTimeRange*)
Sets the send time criterion to the time specified. Use this method to search on a specific date/time; if a range is needed, use `withMessageTimeRange` instead.

Parameters

- **deliveryTimeRange** – the specific time on which to search

Returns this `EmailRecordSearchCriteria` with its `deliveryTimeRange` criterion set to the specified date/time

`withMessageTimeRange`

public `EmailRecordSearchCriteria` **withMessageTimeRange** (`Range<DateTime>` *deliveryTimeRange*)
Sets the sent time criterion to the range specified. Use this method to search on a time range; if searching on a specific date/time is needed, use `withMessageTime` instead.

Parameters

- **deliveryTimeRange** – the date/time range on which to search

Returns this `EmailRecordSearchCriteria` with its `deliveryTimeRange` criterion set to the specified date/time range

`withQueryParams`

public `EmailRecordSearchCriteria` **withQueryParams** (`QueryParams` *queryParams*)
Sets the queryParams of the search criteria to the parameters specified. Use this method when it is necessary to specify order and size of query results. This is used mainly for paging/ordering queries from the UI.

Parameters

- **queryParams** – the query parameters to include with the search criteria

Returns this `EmailRecordSearchCriteria` with its queryParams set to the provided parameters

withSubject

public [EmailRecordSearchCriteria](#) **withSubject** ([String](#) *subject*)

Sets the subject criterion to the subject specified

Parameters

- **subject** – the subject on which to search

Returns this [EmailRecordSearchCriteria](#) with its subject criterion set to the provided subject

withToAddress

public [EmailRecordSearchCriteria](#) **withToAddress** ([String](#) *toAddress*)

Sets the toAddress criterion to the address specified

Parameters

- **toAddress** – the recipient email address on which to search

Returns this [EmailRecordSearchCriteria](#) with its toAddress criterion set to the provided address

7.29 org.motechproject.email.contract

7.29.1 Mail

public class **Mail**

The `Mail` class represents an email message.

Constructors

Mail

public **Mail** ([String](#) *fromAddress*, [String](#) *toAddress*, [String](#) *subject*, [String](#) *message*)

Creates a new instance of `Mail`, with all fields set to the values specified in the parameters.

Parameters

- **fromAddress** – the email address of the sender
- **toAddress** – the email address of the recipient
- **subject** – the subject of the email
- **message** – the body of the email

Methods

equals

public boolean **equals** ([Object](#) *obj*)

Indicates whether some other object is “equal to” this one. Returns true if this `Mail` and the object to compare

have reference equality or their field values are all equal.

Parameters

- **obj** – The reference object with which to compare

Returns true if this object is the same as the obj argument; false otherwise.

getFromAddress

public **String** **getFromAddress** ()

Gets the email address of the sender.

Returns the sender of the message

getMessage

public **String** **getMessage** ()

Gets the message body.

Returns the body of the message

getSubject

public **String** **getSubject** ()

Gets the message subject.

Returns the subject of the message

getText

public **String** **getText** ()

getToAddress

public **String** **getToAddress** ()

Gets the email address of the recipient.

Returns the recipient of the message

hashCode

public int **hashCode** ()

Returns a hash code value for this Mail object.

Returns a hash code value for this Mail object

toString

```
public String toString()
```

Returns a string representation of this Mail object.

Returns a string representation of this Mail object

7.30 org.motechproject.email.domain

7.30.1 DeliveryStatus

```
public enum DeliveryStatus
```

The DeliveryStatus Enum contains the possible delivery states for an email message.

Enum Constants

ERROR

```
public static final DeliveryStatus ERROR
```

There was an error sending the message.

RECEIVED

```
public static final DeliveryStatus RECEIVED
```

The message was received.

SENT

```
public static final DeliveryStatus SENT
```

The message was sent.

7.30.2 EmailRecord

```
public class EmailRecord
```

The EmailRecord class represents a record of a sent Email. This class is exposed as an `org.motechproject.mds.annotations.Entity` through Motech Data Services.

See also: `org.motechproject.mds.annotations`

Constructors

EmailRecord

```
public EmailRecord()
```

Creates a new instance of EmailRecord, with all fields set to null.

EmailRecord

public **EmailRecord** (*String fromAddress, String toAddress, String subject, String message, DateTime deliveryTime, DeliveryStatus deliveryStatus*)

Creates a new instance of `EmailRecord`, with all fields set to the values specified in the parameters.

Parameters

- **fromAddress** – the email address of the sender
- **toAddress** – the email address of the recipient
- **subject** – the subject of the email
- **message** – the body of the email
- **deliveryTime** – the date and time that the email was sent
- **deliveryStatus** – the delivery status of the email

Methods

equals

public boolean **equals** (*Object obj*)

Indicates whether some other object is “equal to” this one. Returns true if this `EmailRecord` and the object to compare have reference equality or their field values are all equal.

Parameters

- **obj** – The reference object with which to compare.

Returns true if this object is the same as the `obj` argument; false otherwise.

getDeliveryStatus

public *DeliveryStatus* **getDeliveryStatus** ()

Gets the delivery status.

Returns the delivery status of the message

getDeliveryTime

public *DateTime* **getDeliveryTime** ()

Gets the delivery time.

Returns the time that the email was sent

getFromAddress

public *String* **getFromAddress** ()

Gets the email address of the sender.

Returns the sender of the message

getId

public Long **getId** ()

getMessage

public String **getMessage** ()

Gets the message body.

Returns the body of the message

getSubject

public String **getSubject** ()

Gets the message subject.

Returns the subject of the message

getToAddress

public String **getToAddress** ()

Gets the email address of the recipient.

Returns the recipient of the message

hashCode

public int **hashCode** ()

Returns a hash code value for this `EmailRecord` object.

Returns a hash code value for this `EmailRecord` object

setFromAddress

public void **setFromAddress** (String *fromAddress*)

Sets the email address of the sender.

Parameters

- **fromAddress** – the sender of the message

setId

public void **setId** (Long *id*)

setMessage

public void **setMessage** (String *message*)

Sets the message body.

Parameters

- **message** – the body of the message

setSubject

public void **setSubject** (*String subject*)

Sets the message subject.

Parameters

- **subject** – the subject of the message

setToAddress

public void **setToAddress** (*String toAddress*)

Sets the email address of the recipient.

Parameters

- **toAddress** – the recipient of the message

toString

public *String* **toString** ()

Returns a string representation of this `EmailRecord` object.

Returns a string representation of this `EmailRecord` object

7.30.3 EmailRecordComparator

public class **EmailRecordComparator** implements `Comparator<EmailRecord>`

The `EmailRecordComparator` class is an implementation of the `Comparator` interface, that allows callers to compare `org.motechproject.email.domain.EmailRecord` objects by a single field.

Constructors

EmailRecordComparator

public **EmailRecordComparator** (*Boolean ascending*, *String compareField*)

Creates a new `EmailRecordComparator` that supports comparison based on the specified field.

Parameters

- **ascending** – boolean indicating whether comparisons should be ascending or descending
- **compareField** – the field for which comparisons should be performed

Methods

compare

public int **compare** (*EmailRecord o1*, *EmailRecord o2*)

Compares its two arguments for order. If `ascending` is `true`, returns a negative integer, zero, or a positive

integer as the first argument is less than, equal to, or greater than the second. If `ascending` is `false`, returns a positive integer, zero, or negative integer as the first argument is less than, equal to, or greater than the second.

Parameters

- **o1** – the first `EmailRecord` to be compared
- **o2** – the second `EmailRecord` to be compared

Returns a positive integer, zero, or negative integer indicating the result of comparing the objects

7.30.4 EmailRecords

public class **EmailRecords**<T>

The `EmailRecords` class wraps the `EmailRecord` list and stores the current item count.

Constructors**EmailRecords**

public **EmailRecords** ()

Creates a new instance of `EmailRecords`, which contains no records.

EmailRecords

public **EmailRecords** (*Integer totalRecords*, *Integer page*, *Integer totalPages*, *List<T> allRecords*)

Creates a new instance of `EmailRecords`, with all fields set to the values specified in the parameters. The `page` and `totalPages` parameters are for the purposes of paginating the list of records in the UI.

Parameters

- **totalRecords** – the total number of records
- **page** – the current page
- **totalPages** – the total number of pages
- **allRecords** – the list of records

Methods**getPage**

public *Integer* **getPage** ()

Gets the current page.

Returns the current page

getRecords

public *Integer* **getRecords** ()

Gets the total number of records.

Returns the total number of records

getRows

public `List<T>` **getRows** ()

Gets the list of records.

Returns the list of records

getTotal

public `Integer` **getTotal** ()

Gets the total number of pages.

Returns the total number of pages

setRows

public void **setRows** (`List<T>` *rows*)

Sets the list of records.

Parameters

- **rows** – the list of records

setTotal

public void **setTotal** (`Integer` *total*)

Sets the total number of records.

Parameters

- **total** – the total number of records

toString

public `String` **toString** ()

Returns a string representation of this `EmailRecords` object.

Returns a string representation of this `EmailRecords` object

7.31 org.motechproject.email.service

7.31.1 EmailAuditService

public interface **EmailAuditService**

The `EmailAuditService` interface provides methods for logging email activity, as well as searching and deleting the email logs.

Methods

countEmailRecords

long **countEmailRecords** ([EmailRecordSearchCriteria](#) *criteria*)

Returns the count of `EmailRecord` entries matching the specified search criteria.

Returns the count of email records matching the provided criteria

delete

void **delete** ([EmailRecord](#) *emailRecord*)

Deletes the specified `EmailRecord` entry from the email log.

findAllEmailRecords

List<[EmailRecord](#)> **findAllEmailRecords** ()

Finds and returns all `EmailRecord` entries in the email log.

Returns all email records in the email log

findById

[EmailRecord](#) **findById** (long *id*)

Finds an `EmailRecord` in the log by ID.

Parameters

- **id** – the identifier of the record to find

Returns the email record that matches the provided identifier, or null if no matching record exists

findEmailRecords

List<[EmailRecord](#)> **findEmailRecords** ([EmailRecordSearchCriteria](#) *criteria*)

Finds and returns all `EmailRecord` entries matching the specified search criteria.

Returns all email records matching the provided criteria

log

void **log** ([EmailRecord](#) *mailRecord*)

Adds the provided `EmailRecord` to the email log.

Parameters

- **mailRecord** – the record to add to the email log

7.31.2 EmailRecordService

public interface **EmailRecordService** extends **MotechDataService**<**EmailRecord**>

This service provides data access for `org.motechproject.email.domain.EmailRecord`. The implementation is generated by Motech Data Services and published as an OSGi service.

Methods

countFind

```
long countFind(String fromAddress, String toAddress, String subject, String message, Range<DateTime>
               deliveryTimeRange, Set<DeliveryStatus> deliveryStatuses)
```

Returns the count of all **EmailRecord** entries matching the specified search parameters.

Parameters

- **fromAddress** – the sender address on which to search
- **toAddress** – the recipient address on which to search
- **subject** – the subject on which to search
- **message** – the message body on which to search
- **deliveryTimeRange** – the delivery time range on which to search
- **deliveryStatuses** – the delivery statuses on which to search

Returns the count of **EmailRecord** entries that match the specified criteria

find

```
List<EmailRecord> find(String fromAddress, String toAddress, String subject, String message,
                      Range<DateTime> deliveryTimeRange, Set<DeliveryStatus> deliveryStatuses,
                      QueryParams queryParams)
```

Finds and returns all **EmailRecord** entries matching the specified search parameters. This method is exposed as a `org.motechproject.mds.annotations.Lookup` through Motech Data Services.

Parameters

- **fromAddress** – the sender address on which to search
- **toAddress** – the recipient address on which to search
- **subject** – the subject on which to search
- **message** – the message body on which to search
- **deliveryTimeRange** – the delivery time range on which to search
- **deliveryStatuses** – the delivery statuses on which to search
- **queryParams** – the query parameters to include with the search criteria

Returns the list of **EmailRecord** entries that match the specified criteria

See also: `org.motechproject.mds.annotations`

findByRecipientAddress

List<EmailRecord> **findByRecipientAddress** (String *recipientAddress*)

Finds and returns all EmailRecord entries for the specified recipient address. This method is exposed as a `org.motechproject.mds.annotations.Lookup` through Motech Data Services.

Parameters

- **recipientAddress** – the recipient address on which to search

Returns the list of EmailRecord entries that match the specified address

See also: `org.motechproject.mds.annotations`

7.31.3 EmailSenderService

public interface **EmailSenderService**

The EmailSenderService interface provides a method for sending email.

Methods

send

void **send** (Mail *message*)

Attempts to send the supplied email message. Adds an `org.motechproject.email.domain.EmailRecord` entry to the log with the details of the activity.

Parameters

- **message** – the message to send

Throws

- **MailException** – if an error occurs when attempting to send the message

7.32 org.motechproject.event

7.32.1 MotechEvent

public class **MotechEvent** implements `Serializable`

Motech Scheduled Event data carrier class, Instance of this class with event specific data will be send by Motech Scheduler when a scheduled event is fired

This class is immutable

Fields

EVENT_TYPE_KEY_NAME

public static final String **EVENT_TYPE_KEY_NAME**

PARAM_DISCARDED_MOTECH_EVENT

```
public static final String PARAM_DISCARDED_MOTECH_EVENT
```

PARAM_INVALID_MOTECH_EVENT

```
public static final String PARAM_INVALID_MOTECH_EVENT
```

PARAM_REDELIVERY_COUNT

```
public static final String PARAM_REDELIVERY_COUNT
```

Constructors**MotechEvent**

```
public MotechEvent ()
```

MotechEvent

```
public MotechEvent (String subject)
```

Constructor with subject only (parameters can be added interactively)

Parameters

- **subject** –
 - event destination

Throws

- **IllegalArgumentException** –

MotechEvent

```
public MotechEvent (String subject, Map<String, Object> parameters)
```

Constructor

Parameters

- **subject** –
 - event type: Pill Reminder, Appointment Reminder ...
- **parameters** –
 - a Map of additional parameters

Throws

- **IllegalArgumentException** –

Methods

copy

public [MotechEvent](#) **copy** ([String](#) *subject*, [Map](#)<[String](#), [Object](#)> *parameters*)

equals

public boolean **equals** ([Object](#) *o*)

getEndTime

public [Date](#) **getEndTime** ()

getId

public [UUID](#) **getId** ()

getMessageRedeliveryCount

public int **getMessageRedeliveryCount** ()

getParameters

public [Map](#)<[String](#), [Object](#)> **getParameters** ()
Sets empty [HashMap](#) if parameters=null

getSubject

public [String](#) **getSubject** ()

hashCode

public int **hashCode** ()

incrementMessageRedeliveryCount

public void **incrementMessageRedeliveryCount** ()

isLastEvent

public boolean **isLastEvent** ()

setEndTime

```
public MotechEvent setEndTime (Date endDate)
```

setId

```
public void setId (UUID id)
```

setLastEvent

```
public MotechEvent setLastEvent (boolean lastEvent)
```

toString

```
public String toString ()
```

7.33 org.motechproject.event.listener

7.33.1 EventListener

```
public interface EventListener
```

Methods

getIdentifier

```
String getIdentifier ()
```

Retrieve a unique identifier/key for the given listener class. This identifier is used when messages are destined for this specific listener type

Returns Unique listener identifier/key

handle

```
void handle (MotechEvent event)
```

Handle an particular event that has been received

Parameters

- **event** –

7.33.2 EventListenerRegistryService

```
public interface EventListenerRegistryService
```

This interface is necessary for OSGi service publication. The implementing class acts as a registry for all scheduled event listeners. One can register themselves to listen for a specific set of event types.

Methods

clearListenersForBean

void **clearListenersForBean** (*String* *beanName*)

This method is responsible for removing listeners for a particular bean. This is necessary when bundles are stopped in some fashion so that the listener does not persist.

Parameters

- **beanName** – The bean name from the Spring context of the candidate class for listener clearing

getListenerCount

int **getListenerCount** (*String* *subject*)

Get the count of listeners for a particular subject.

Parameters

- **subject** –

getListeners

Set<EventListener> **getListeners** (*String* *subject*)

Retrieve a list of event listeners for a given event type. If there are no listeners, an empty list is returned.

Parameters

- **subject** – The event type that you are seeking listeners for

Returns A list of scheduled event listeners that are interested in that event

hasListener

boolean **hasListener** (*String* *subject*)

See if a particular subject has any listeners.

Parameters

- **subject** –

registerListener

void **registerListener** (*EventListener* *listener*, List<*String*> *subjects*)

Register an event listener to be notified when events of a given type are received via the Server JMS Event Queue.

Parameters

- **listener** – the listener instance
- **subjects** – the event types that a listener is interested in

registerListener

void **registerListener** (*EventListener listener*, *String subject*)

7.33.3 EventRelay

public interface **EventRelay**

Methods

sendEventMessage

void **sendEventMessage** (*MotechEvent motechEvent*)

7.34 org.motechproject.event.listener.annotations

7.34.1 MotechListener

public @interface **MotechListener**

Author yyonkov

7.34.2 MotechListenerAbstractProxy

public abstract class **MotechListenerAbstractProxy** implements *EventListener*
Event Listener Proxy base abstract class

Author yyonkov

Constructors

MotechListenerAbstractProxy

public **MotechListenerAbstractProxy** (*String name*, *Object bean*, *Method method*)

Parameters

- **name** –
- **bean** –
- **method** –

Methods

callHandler

public abstract void **callHandler** (*MotechEvent event*)
Needs to be implemented by concrete Proxies

Parameters

- **event** –

getBean

```
public Object getBean ()
```

getIdentifier

```
public String getIdentifier ()
```

getMethod

```
public Method getMethod ()
```

handle

```
public void handle (MotechEvent event)
```

7.34.3 MotechListenerEventProxy

```
public class MotechListenerEventProxy extends MotechListenerAbstractProxy
```

Responsible for dispatching to void doSomething (MotechEvent event) {} type of handlers

Author yyonkov

Constructors**MotechListenerEventProxy**

```
public MotechListenerEventProxy (String name, Object bean, Method method)
```

Parameters

- **name** –
- **bean** –
- **method** –

Methods**callHandler**

```
public void callHandler (MotechEvent event)
```


7.34.4 MotechListenerNamedParametersProxy

public class **MotechListenerNamedParametersProxy** extends [MotechListenerAbstractProxy](#)

Author yyonkov

Constructors

MotechListenerNamedParametersProxy

public **MotechListenerNamedParametersProxy** ([String name](#), [Object bean](#), [Method method](#))

Parameters

- **name** –
- **bean** –
- **method** –

Methods

callHandler

public void **callHandler** ([MotechEvent event](#))

7.34.5 MotechListenerType

public enum **MotechListenerType**

ENUM defining the annotation types

Author yyonkov

Enum Constants

MOTECH_EVENT

public static final [MotechListenerType](#) **MOTECH_EVENT**

NAMED_PARAMETERS

public static final [MotechListenerType](#) **NAMED_PARAMETERS**

7.34.6 MotechParam

public @interface **MotechParam**

Author yyonkov

7.35 org.motechproject.event.queue

7.35.1 MotechCachingConnectionFactory

public class **MotechCachingConnectionFactory** extends [CachingConnectionFactory](#)
The `MotechCachingConnectionFactory` is used to created connection to ActiveMQ.

Methods

doCreateConnection

protected [Connection](#) **doCreateConnection** ()

setBrokerUrl

public void **setBrokerUrl** ([String](#) *brokerURL*)

setPassword

public void **setPassword** ([String](#) *password*)

setUsername

public void **setUsername** ([String](#) *username*)

7.35.2 MotechEventConfig

public class **MotechEventConfig**

Methods

getMessageMaxRedeliveryCount

public int **getMessageMaxRedeliveryCount** ()

getMessageRedeliveryDelay

public long **getMessageRedeliveryDelay** ()

7.35.3 MotechEventHeaderMapper

public class **MotechEventHeaderMapper** extends [DefaultJmsHeaderMapper](#)
For the delay to work, set attribute `schedulerSupport="true"` in the broker element of the `activemq.xml` Ref:
<http://activemq.apache.org/delay-and-schedule-message-delivery.html>

Methods

fromHeaders

public void **fromHeaders** ([MessageHeaders](#) *messageHeaders*, [Message](#) *message*)

7.35.4 MotechEventTransformer

public class **MotechEventTransformer**

Methods

transform

public [MotechEvent](#) **transform** ([MotechEvent](#) *motechEvent*)

7.35.5 OutboundEventGateway

public interface **OutboundEventGateway**

Methods

sendEventMessage

void **sendEventMessage** ([MotechEvent](#) *motechEvent*)

Sends the given MotechEvent message as a payload to the message channel defined in the Spring Integration configuration file.

Parameters

- **motechEvent** –

7.36 org.motechproject.mds.annotations

7.36.1 Cascade

public @interface **Cascade**

The `Cascade` annotation is used by developers to set correct cascade properties for the given field that is a relationship.

See also: `org.motechproject.mds.annotations.internal.FieldProcessor`

7.36.2 Entity

public @interface **Entity**

The `Entity` annotation is used by developers to point classes, that should be mapped as Motech Dataservices Entities. The discovery logic for this annotation is done in `org.motechproject.mds.annotations.internal.EntityProcessor`

See also: `org.motechproject.mds.annotations.internal.EntityProcessor`

7.36.3 Field

public @interface **Field**

The `Field` annotation is used by developers to point fields, that should be mapped as entity fields. The discovery logic for this annotation is done in `org.motechproject.mds.annotations.internal.FieldProcessor`.

Only fields, 'getter' or 'setter' methods can have this annotation for other methods this annotation is omitted.

See also: `org.motechproject.mds.annotations.internal.FieldProcessor`

7.36.4 Ignore

public @interface **Ignore**

Thanks to this annotation, developers can point public fields and getter/setter method of a private field that should not be added to the schema definition of an entity. By default, all public fields of an object are added. The discovery logic for this annotation is done in the `FieldProcessor`

See also: `org.motechproject.mds.annotations.internal.FieldProcessor`

7.36.5 InSet

public @interface **InSet**

The annotated element must have value that will be in defined set.

Supported types are:

- Integer
- Double
- int, double

7.36.6 Lookup

public @interface **Lookup**

The `Lookup` annotation is used by developers to point methods, in classes that implements `org.motechproject.mds.service.MotechDataService`, that should be mapped as Motech Dataservices Lookups. The discovery logic for this annotation is done in `org.motechproject.mds.annotations.internal.LookupProcessor`

See also: `org.motechproject.mds.annotations.internal.LookupProcessor`

7.36.7 LookupField

public @interface **LookupField**

The `LookupField` annotation allows developers to point fields in `Lookup` method, that should be mapped as `Lookup` fields for Developer Defined `Lookup`. The discovery logic for this annotation is done in `org.motechproject.mds.annotations.internal.LookupProcessor`

See also: `org.motechproject.mds.annotations.internal.LookupProcessor`

7.36.8 NotInSet

public @interface **NotInSet**

The annotated element must not have value that will be in defined set.

Supported types are:

- Integer
- Double
- int, double

7.36.9 UIDisplayable

public @interface **UIDisplayable**

The `UIDisplayable` annotation is used by developers to mark a field as being in the default display for a listing of objects. The discovery logic for this annotation is done in `org.motechproject.mds.annotations.internal.UIDisplayableProcessor`.

Only fields, 'getter' or 'setter' methods can have this annotation for other methods this annotation is omitted.

See also: `org.motechproject.mds.annotations.internal.UIDisplayableProcessor`

7.36.10 UIFilterable

public @interface **UIFilterable**

The `UIFilterable` annotation is used by developers to mark a field that allow users to filter a list of objects by the values in the field. The discovery logic for this annotation is done in `org.motechproject.mds.annotations.internal.UIFilterableProcessor`.

Only fields, 'getter' or 'setter' methods can have this annotation for other methods this annotation is omitted. Also this annotation is permitted on fields of type Date, Boolean or List.

See also: `org.motechproject.mds.annotations.internal.UIFilterableProcessor`

7.37 org.motechproject.mds.builder

7.37.1 EntityBuilder

public interface **EntityBuilder**

An entity builder is responsible for building the entity class from an Entity schema.

Methods

build

`ClassData` **build** (*Entity entity*)

Builds a class definition for a given entity. The class is not registered with any classloader.

Parameters

- **entity** – the entity schema

Returns bytes of the newly constructed class

buildDDE

ClassData **buildDDE** (*Entity entity*, *Bundle bundle*)

Builds Developer Defined Entity. The main difference between regular build method and this one, is that this method fetches the class definition from the given bundle, and injects members to the constructed class from there, if possible, rather than building everything from scratch.

Parameters

- **entity** – the entity schema
- **bundle** – the bundle to fetch class definition from

Returns bytes of the constructed class

buildHistory

ClassData **buildHistory** (*Entity entity*)

Builds History class definition for the given entity. The history class definition contains the same members as the entity class, plus some fields history-exclusive, like schema version.

Parameters

- **entity** – the entity schema

Returns bytes of the constructed class

buildTrash

ClassData **buildTrash** (*Entity entity*)

Builds Trash class definition for the given entity. The trash class definition contains the same members as the entity class, plus some fields trash-exclusive.

Parameters

- **entity** – the entity schema

Returns bytes of the constructed class

prepareHistoryClass

void **prepareHistoryClass** (*Entity entity*)

Builds empty history class definition for the given entity and adds it to the class pool.

Parameters

- **entity** – the entity schema

prepareTrashClass

void **prepareTrashClass** (*Entity entity*)

Builds empty trash class definition for the given entity and adds it to the class pool.

Parameters

- **entity** – the entity schema

7.37.2 EntityInfrastructureBuilder

public interface **EntityInfrastructureBuilder**

The `EntityInfrastructureBuilder` is responsible for building infrastructure for a given entity: repository, interface and service classes.

Methods

buildHistoryInfrastructure

`List<ClassData> buildHistoryInfrastructure (String className)`

Builds the repository, interface and implementation of this interface classes for the given history class. The names for classes are generated by `org.motechproject.mds.util.ClassName.getRepositoryName (String)`, `org.motechproject.mds.util.ClassName.getInterfaceName (String)`, `org.motechproject.mds.util.ClassName.getServiceName (String)`, respectively.

Parameters

- **className** – a name of history class.

Returns a list of classes that represents infrastructure for the history class.

buildInfrastructure

`List<ClassData> buildInfrastructure (Entity entity)`

Builds the repository, interface and implementation of this interface classes for the given entity. The names for classes are generated by `org.motechproject.mds.util.ClassName.getRepositoryName (String)`, `org.motechproject.mds.util.ClassName.getInterfaceName (String)`, `org.motechproject.mds.util.ClassName.getServiceName (String)`, respectively.

Parameters

- **entity** – an instance of `org.motechproject.mds.domain.Entity`

Returns a list of classes that represents infrastructure for the given entity.

7.37.3 EntityMetadataBuilder

public interface **EntityMetadataBuilder**

The `EntityMetadataBuilderImpl` class is responsible for building jdo metadata for an entity class.

Methods

addBaseMetadata

`void addBaseMetadata (JDOMetadata jdoMetadata, ClassData classData)`

Adds base information about package and class name to a `javax.jdo.metadata.JDOMetadata` instance.

Parameters

- **jdoMetadata** – an empty instance of `javax.jdo.metadata.JDOMetadata`.

- **classData** – an instance of `org.motechproject.mds.domain.ClassData`

addEntityMetadata

void **addEntityMetadata** (`JDOMetadata jdoMetadata`, `Entity entity`)

Adds information about package and class name to a `javax.jdo.metadata.JDOMetadata` instance.

Parameters

- **jdoMetadata** – a empty instance of `javax.jdo.metadata.JDOMetadata`.
- **entity** – a instance of `org.motechproject.mds.domain.Entity`

addHelperClassMetadata

void **addHelperClassMetadata** (`JDOMetadata jdoMetadata`, `ClassData classData`, `Entity entity`)

Creates metadata with basic information about package and class name to the `javax.jdo.metadata.JDOMetadata` instance. Additionally, fetches fields from passed entities and adds metadata for fields, if it's necessary. If entity is null, it will work just like `addBaseMetadata(JDOMetadata, ClassData)` and won't add any metadata for fields.

Parameters

- **jdoMetadata** – an empty instance of `javax.jdo.metadata.JDOMetadata`.
- **classData** – an instance of `org.motechproject.mds.domain.ClassData`
- **entity** – an entity to fetch fields from

fixEnhancerIssuesInMetadata

void **fixEnhancerIssuesInMetadata** (`JDOMetadata jdoMetadata`)

This updates the metadata after enhancement. Nucleus makes some “corrections” which do not work with the runtime enhancer.

7.37.4 EnumBuilder

public interface **EnumBuilder**

An enum builder is responsible for building the enum class with the same values as those are defined in the field.

Methods

build

`ClassData` **build** (`ComboBoxHolder holder`)

7.37.5 MDSConstruktor

public interface **MDSConstruktor**

This interface provide method to create a class for the given entity. The implementation of this interface should also construct other classes like repository, service interface and implementation for this service interface.

Methods

constructEntities

boolean **constructEntities** (boolean *buildDDE*)

Creates a class definition and insert it into the MDS class loader based on data from database. The implementation of this method should also create a repository, interface (when it's necessary) and implementation of this interface.

After executing this method, there should be possible to create an instance of the given class definition and save it to database by `javax.jdo.PersistenceManager` provided by DataNucleus.

An interface related with class definition should be created only for entities from outside bundles and if the bundle does not define its interface.

Parameters

- **buildDDE** – true if class definitions for entities from outside bundles should also be created; otherwise false.

Returns true if there were entities for which class definitions should be created; otherwise false.

updateFields

void **updateFields** (Long *entityId*, Map<String, String> *fieldNameChanges*)

Updates the field names of an entity. This method changes the database schema by changing column names to the new value. This is done for the entity instances, history instances and trash instances.

Parameters

- **entityId** – The ID of an entity to update
- **fieldNameChanges** – A map, indexed by current field names and values being updated field names.

7.37.6 MDSDDataProviderBuilder

public interface **MDSDDataProviderBuilder**

Methods

generateDataProvider

String **generateDataProvider** ()

7.38 org.motechproject.mds.config

7.38.1 DeleteMode

public enum **DeleteMode**

The `DeleteMode` presents what should happen with objects when they are deleted. They can be deleted permanently `DELETE` or moved to the trash `TRASH`. This enum is related with the property `org.motechproject.mds.util.Constants.Config.MDS_DELETE_MODE`.

The `UNKNOWN` value should not be used in code as appropriate value. It was added to ensure that the `fromString(String)` method will not return `{@value null}` value.

Enum Constants

DELETE

```
public static final DeleteMode DELETE
```

TRASH

```
public static final DeleteMode TRASH
```

UNKNOWN

```
public static final DeleteMode UNKNOWN
```

7.38.2 MdsConfig

```
public class MdsConfig
```

Class responsible for handling MDS configuration. Since MDS don't use Server Config everything connected to the MDS configuration needs to be handled by the module itself.

Constructors

MdsConfig

```
public MdsConfig ()
```

Methods

asProperties

```
public Properties asProperties ()
```

getDataNucleusProperties

```
public Properties getDataNucleusProperties ()
```

getProperties

```
public Properties getProperties (String filename)
```

getResourceFileName

```
public String getResourceFileName (Resource resource)
```

setConfig

```
public void setConfig (List<Resource> resources)
```

7.38.3 ModuleSettings

```
public class ModuleSettings extends Properties
```

The `ModuleSettings` contains the base module settings which are inside the `org.motechproject.mds.util.Constants.Config.MODULE_FILE`. The getters and setters inside this class always checks the given property and if it is incorrect then the default value of the given property will be returned.

Fields

DEFAULT_DELETE_MODE

```
public static final DeleteMode DEFAULT_DELETE_MODE
```

DEFAULT_EMPTY_TRASH

```
public static final Boolean DEFAULT_EMPTY_TRASH
```

DEFAULT_TIME_UNIT

```
public static final TimeUnit DEFAULT_TIME_UNIT
```

DEFAULT_TIME_VALUE

```
public static final Integer DEFAULT_TIME_VALUE
```

Methods

equals

```
public boolean equals (Object obj)
```

getDeleteMode

```
public DeleteMode getDeleteMode ()
```

getTimeUnit

```
public TimeUnit getTimeUnit ()
```

getTimeValue

public [Integer](#) **getTimeValue** ()

hashCode

public int **hashCode** ()

isEmptyTrash

public [Boolean](#) **isEmptyTrash** ()

setDeleteMode

public void **setDeleteMode** ([String](#) *deleteMode*)

setDeleteMode

public void **setDeleteMode** ([DeleteMode](#) *deleteMode*)

setEmptyTrash

public void **setEmptyTrash** ([String](#) *emptyTrash*)

setEmptyTrash

public void **setEmptyTrash** ([Boolean](#) *emptyTrash*)

setTimeUnit

public void **setTimeUnit** ([String](#) *timeUnit*)

setTimeUnit

public void **setTimeUnit** ([TimeUnit](#) *timeUnit*)

setTimeValue

public void **setTimeValue** ([String](#) *timeValue*)

setTimeValue

public void **setTimeValue** ([Integer](#) *timeValue*)

`toString`

```
public String toString ()
```

7.38.4 ModuleSettings.Deserializer

```
public static final class Deserializer extends JsonSerializer<ModuleSettings>
```

Methods

deserialize

```
public ModuleSettings deserialize (JsonParser jp, DeserializationContext ctx)
```

7.38.5 ModuleSettings.Serializer

```
public static final class Serializer extends JsonSerializer<ModuleSettings>
```

Methods

serialize

```
public void serialize (ModuleSettings value, JsonGenerator jgen, SerializerProvider provider)
```

7.38.6 SettingsService

```
public interface SettingsService
```

The `SettingsService` is a service class for `org.motechproject.server.config.SettingsFacade`. Its main purpose is to create better access to module settings for developers.

Methods

getDeleteMode

```
DeleteMode getDeleteMode ()
```

getModuleSettings

```
ModuleSettings getModuleSettings ()
```

getProperties

```
Properties getProperties ()
```

getTimeUnit

```
TimeUnit getTimeUnit ()
```

getTimeValue

```
Integer getTimeValue ()
```

isEmptyTrash

```
Boolean isEmptyTrash ()
```

saveModuleSettings

```
void saveModuleSettings (ModuleSettings settings)
```

7.38.7 TimeUnit

```
public enum TimeUnit
```

The `TimeUnit` specifies what time unit should be used to specify time when the module trash should be cleaned. This enum is related with the property `org.motechproject.mds.util.Constants.Config.MDS_TIME_UNIT`.

Each value from this enum can be converted to long value that presents time interval in milliseconds. For example the `HOURS` value is equal to `{@value 3.6E6}`.

The `UNKNOWN` value should not be used in code as appropriate value. It was added to ensure that the `fromString(String)` method will not return `{@value null}` value.

Enum Constants

DAYS

```
public static final TimeUnit DAYS
```

HOURS

```
public static final TimeUnit HOURS
```

MONTHS

```
public static final TimeUnit MONTHS
```

UNKNOWN

```
public static final TimeUnit UNKNOWN
```

WEEKS

public static final [TimeUnit](#) **WEEKS**

YEARS

public static final [TimeUnit](#) **YEARS**

7.39 org.motechproject.mds.domain

7.39.1 BrowsingSettings

public class **BrowsingSettings**

The `BrowsingSettings` contains information about fields that will be visible on UI and could be used as filter on UI.

This class is read only (the data are not saved to database) and its main purpose is to provide methods that help developer to get displayed and filterable fields.

Constructors

BrowsingSettings

public **BrowsingSettings** ([Entity](#) *entity*)

Methods

getDisplayedFields

public [List](#)<[Field](#)> **getDisplayedFields** ()

getEntity

public [Entity](#) **getEntity** ()

getFilterableFields

public [List](#)<[Field](#)> **getFilterableFields** ()

setEntity

public void **setEntity** ([Entity](#) *entity*)

toDto

public [BrowsingSettingsDto](#) **toDto** ()

7.39.2 ClassData

public class **ClassData**

Represents a class name and its byte code.

Constructors

ClassData

public **ClassData** (*String* *className*, byte[] *bytecode*)

ClassData

public **ClassData** (*String* *className*, byte[] *bytecode*, boolean *interfaceClass*)

ClassData

public **ClassData** (*Entity* *entity*, byte[] *bytecode*)

ClassData

public **ClassData** (*Entity* *entity*, byte[] *bytecode*, boolean *interfaceClass*)

ClassData

public **ClassData** (*String* *className*, *String* *module*, *String* *namespace*, byte[] *bytecode*)

ClassData

public **ClassData** (*String* *className*, *String* *module*, *String* *namespace*, byte[] *bytecode*, *EntityType* *type*)

ClassData

public **ClassData** (*String* *className*, *String* *module*, *String* *namespace*, byte[] *bytecode*, boolean *interfaceClass*)

ClassData

public **ClassData** (*String* *className*, *String* *module*, *String* *namespace*, byte[] *bytecode*, boolean *interfaceClass*, *EntityType* *type*)

Methods

getBytecode

public byte[] **getBytecode** ()

getClassName

```
public String getClassName ()
```

getModule

```
public String getModule ()
```

getNamespace

```
public String getNamespace ()
```

getType

```
public EntityType getType ()
```

isDDE

```
public boolean isDDE ()
```

isInterfaceClass

```
public boolean isInterfaceClass ()
```

toString

```
public String toString ()
```

7.39.3 ComboboxHolder

```
public class ComboboxHolder extends FieldHolder
```

The main purpose of this class is to find out what kind of type should be used when the field will be added to the class definition.

Constructors**ComboboxHolder**

```
public ComboboxHolder (Field field)
```

ComboboxHolder

```
public ComboboxHolder (Entity entity, Field field)
```

ComboboxHolder

```
public ComboboxHolder (EntityDto entity, FieldDto field)
```

ComboboxHolder

```
public ComboboxHolder (List<? extends Pair<String, String>> metadata, List<? extends Pair<String, ?>>  
                      settings, String defaultEnumName)
```

Methods

getEnumName

```
public String getEnumName ()
```

getValues

```
public String[] getValues ()
```

isAllowMultipleSelections

```
public boolean isAllowMultipleSelections ()
```

isAllowUserSupplied

```
public boolean isAllowUserSupplied ()
```

isEnum

```
public boolean isEnum ()
```

isEnumList

```
public boolean isEnumList ()
```

isString

```
public boolean isString ()
```

isStringList

```
public boolean isStringList ()
```

7.39.4 ConfigSettings

public class **ConfigSettings**

Constructors

ConfigSettings

public **ConfigSettings** ()

ConfigSettings

public **ConfigSettings** ([DeleteMode](#) *deleteMode*, boolean *emptyTrash*, int *afterTimeValue*, [TimeUnit](#) *afterTimeUnit*)

Methods

getAfterTimeUnit

public [TimeUnit](#) **getAfterTimeUnit** ()

getAfterTimeValue

public int **getAfterTimeValue** ()

getDeleteMode

public [DeleteMode](#) **getDeleteMode** ()

getEmptyTrash

public boolean **getEmptyTrash** ()

getId

public [Long](#) **getId** ()

setAfterTimeUnit

public void **setAfterTimeUnit** ([TimeUnit](#) *afterTimeUnit*)

setAfterTimeValue

public void **setAfterTimeValue** (int *afterTimeValue*)

setDeleteMode

```
public void setDeleteMode (DeleteMode deleteMode)
```

setEmptyTrash

```
public void setEmptyTrash (boolean emptyTrash)
```

setId

```
public void setId (Long id)
```

7.39.5 Entity

```
public class Entity
```

The `Entity` class contains information about an entity. Also it contains information about advanced settings related with the entity.

Constructors**Entity**

```
public Entity ()
```

Entity

```
public Entity (String className)
```

Entity

```
public Entity (String className, String module, String namespace, SecurityMode securityMode)
```

Entity

```
public Entity (String className, String name, String module, String namespace, SecurityMode security-  
Mode, Set<String> securityMembers)
```

Methods**addField**

```
public void addField (Field field)
```

addLookup

```
public void addLookup (Lookup lookup)
```

advancedSettingsDto

```
public AdvancedSettingsDto advancedSettingsDto ()
```

getBrowsingSettings

```
public BrowsingSettings getBrowsingSettings ()
```

getClassName

```
public String getClassName ()
```

getDrafts

```
public List<EntityDraft> getDrafts ()
```

getEntityVersion

```
public Long getEntityVersion ()
```

getField

```
public Field getField (Long id)
```

getField

```
public Field getField (String name)
```

getFields

```
public List<Field> getFields ()
```

getId

```
public Long getId ()
```

getLookupById

```
public Lookup getLookupById (Long lookupId)
```

getLookupByName

public [Lookup](#) **getLookupByName** ([String](#) *lookupName*)

getLookups

public [List](#)<[Lookup](#)> **getLookups** ()

getLookupsDtos

public [List](#)<[LookupDto](#)> **getLookupsDtos** ()

getModule

public [String](#) **getModule** ()

getName

public [String](#) **getName** ()

getNamespace

public [String](#) **getNamespace** ()

getRestOptions

public [RestOptions](#) **getRestOptions** ()

getSecurityMembers

public [Set](#)<[String](#)> **getSecurityMembers** ()

getSecurityMode

public [SecurityMode](#) **getSecurityMode** ()

getStringComboboxFields

public [List](#)<[Field](#)> **getStringComboboxFields** ()

getSuperClass

public [String](#) **getSuperClass** ()

getTracking

```
public Tracking getTracking ()
```

isAbstractClass

```
public boolean isAbstractClass ()
```

isActualEntity

```
public boolean isActualEntity ()
```

isBaseEntity

```
public boolean isBaseEntity ()
```

isDDE

```
public boolean isDDE ()
```

isDraft

```
public boolean isDraft ()
```

removeField

```
public void removeField (Long fieldId)
```

removeLookup

```
public void removeLookup (Long lookupId)
```

setAbstractClass

```
public void setAbstractClass (boolean abstractClass)
```

setClassName

```
public void setClassName (String className)
```

setDrafts

```
public void setDrafts (List<EntityDraft> drafts)
```

setEntityVersion

public void **setEntityVersion** (*Long entityVersion*)

setFields

public void **setFields** (*List<Field> fields*)

setId

public void **setId** (*Long id*)

setLookups

public void **setLookups** (*List<Lookup> lookups*)

setModule

public void **setModule** (*String module*)

setName

public final void **setName** (*String name*)

setNamespace

public void **setNamespace** (*String namespace*)

setRestOptions

public void **setRestOptions** (*RestOptions restOptions*)

setSecurity

public void **setSecurity** (*SecurityMode securityMode, List<String> securityMembersList*)

setSecurityMembers

public void **setSecurityMembers** (*Set<String> securityMembers*)

setSecurityMode

public void **setSecurityMode** (*SecurityMode securityMode*)

setSuperClass

```
public void setSuperClass (String superClass)
```

setTracking

```
public void setTracking (Tracking tracking)
```

toDto

```
public EntityDto toDto ()
```

updateAdvancedSetting

```
public void updateAdvancedSetting (AdvancedSettingsDto advancedSettings)
```

updateFromDraft

```
public void updateFromDraft (EntityDraft draft)
```

7.39.6 EntityAudit

```
public class EntityAudit extends Entity
```

This class represents a single historical commit of an Entity.

Methods**getModificationDate**

```
public DateTime getModificationDate ()
```

getOwnerUsername

```
public String getOwnerUsername ()
```

getVersion

```
public Long getVersion ()
```

isActualEntity

```
public boolean isActualEntity ()
```

setModificationDate

```
public void setModificationDate (DateTime modificationDate)
```

setOwnerUsername

```
public void setOwnerUsername (String ownerUsername)
```

setVersion

```
public void setVersion (Long version)
```

7.39.7 EntityDraft

```
public class EntityDraft extends Entity
```

This class represents a users draft of an Entity. A draft is a users work in progress from the UI. This shares the table with its superclass, *Entity*.

Constructors**EntityDraft**

```
public EntityDraft ()
```

Methods**getDraftOwnerUsername**

```
public String getDraftOwnerUsername ()
```

getFieldNameChanges

```
public Map<String, String> getFieldNameChanges ()
```

getLastModificationDate

```
public DateTime getLastModificationDate ()
```

getParentEntity

```
public Entity getParentEntity ()
```

getParentVersion

```
public Long getParentVersion ()
```

isActualEntity

public boolean **isActualEntity** ()

isChangesMade

public boolean **isChangesMade** ()

isDraft

public boolean **isDraft** ()

isOutdated

public boolean **isOutdated** ()

setChangesMade

public void **setChangesMade** (boolean *changesMade*)

setDraftOwnerUsername

public void **setDraftOwnerUsername** ([String](#) *draftOwnerUsername*)

setFieldNameChanges

public void **setFieldNameChanges** ([Map](#)<[String](#), [String](#)> *fieldNameChanges*)

setLastModificationDate

public void **setLastModificationDate** ([DateTime](#) *lastModificationDate*)

setParentEntity

public void **setParentEntity** ([Entity](#) *parentEntity*)

setParentVersion

public void **setParentVersion** ([Long](#) *parentVersion*)

toDto

public [EntityDto](#) **toDto** ()

7.39.8 EntityInfo

public class **EntityInfo**

The `EntityInfo` class contains base information about the given entity like class name or infrastructure classes name.

See also: `org.motechproject.mds.service.JarGeneratorService`

Methods

getClassName

public `String` **getClassName** ()

getInfrastructure

public `String[]` **getInfrastructure** ()

getInterfaceName

public `String` **getInterfaceName** ()

getName

public `String` **getName** ()

getRepository

public `String` **getRepository** ()

getServiceName

public `String` **getServiceName** ()

setClassName

public void **setClassName** (`String` *className*)

setInterfaceName

public void **setInterfaceName** (`String` *interfaceName*)

setRepository

public void **setRepository** (`String` *repository*)

`setServiceName`

```
public void setServiceName (String serviceName)
```

7.39.9 EntityType

```
public enum EntityType
```

Enum Constants

HISTORY

```
public static final EntityType HISTORY
```

STANDARD

```
public static final EntityType STANDARD
```

TRASH

```
public static final EntityType TRASH
```

7.39.10 Field

```
public class Field
```

The `Field` class contains information about a single field.

Constructors

Field

```
public Field ()
```

Field

```
public Field (Entity entity, String displayName, String name, Set<Lookup> lookups)
```

Field

```
public Field (Entity entity, String name, String displayName, Type type, boolean required, boolean readOnly)
```

Field

```
public Field (Entity entity, String displayName, String name, boolean required, boolean readOnly, String  
    defaultValue, String tooltip, Set<Lookup> lookups)
```

Methods

addMetadata

public void **addMetadata** ([FieldMetadata](#) *metadata*)

addSetting

public void **addSetting** ([FieldSetting](#) *setting*)

addValidation

public void **addValidation** ([FieldValidation](#) *validation*)

copy

public [Field](#) **copy** ()

getDefaultValue

public [String](#) **getDefaultValue** ()

getDisplayName

public [String](#) **getDisplayName** ()

getEntity

public [Entity](#) **getEntity** ()

getId

public [Long](#) **getId** ()

getLookups

public [Set](#)<[Lookup](#)> **getLookups** ()

getMetadata

public [List](#)<[FieldMetadata](#)> **getMetadata** ()

getMetadata

public [FieldMetadata](#) **getMetadata** ([String](#) *key*)

getMetadataById

```
public FieldMetadata getMetadataById (Long id)
```

getName

```
public String getName ()
```

getSettingByName

```
public FieldSetting getSettingByName (String name)
```

getSettings

```
public List<FieldSetting> getSettings ()
```

getTooltip

```
public String getTooltip ()
```

getType

```
public Type getType ()
```

getUIDisplayPosition

```
public Long getUIDisplayPosition ()
```

getValidationByName

```
public FieldValidation getValidationByName (String name)
```

getValidations

```
public List<FieldValidation> getValidations ()
```

isExposedViaRest

```
public boolean isExposedViaRest ()
```

isReadOnly

```
public boolean isReadOnly ()
```

isRequired

public boolean **isRequired**()

isTracked

public boolean **isTracked**()

isUIDisplayable

public boolean **isUIDisplayable**()

isUIFilterable

public boolean **isUIFilterable**()

setDefaultValue

public void **setDefaultValue**(String *defaultValue*)

setDisplayName

public void **setDisplayName**(String *displayName*)

setEntity

public void **setEntity**(Entity *entity*)

setExposedViaRest

public void **setExposedViaRest**(boolean *exposedViaRest*)

setId

public void **setId**(Long *id*)

setLookups

public void **setLookups**(Set<Lookup> *lookups*)

setMetadata

public void **setMetadata**(List<FieldMetadata> *metadata*)

setName

public final void **setName** (*String name*)

setReadOnly

public void **setReadOnly** (boolean *readOnly*)

setRequired

public void **setRequired** (boolean *required*)

setSettings

public void **setSettings** (*List<FieldSetting> settings*)

setTooltip

public void **setTooltip** (*String tooltip*)

setTracked

public void **setTracked** (boolean *tracked*)

setType

public void **setType** (*Type type*)

setUIDisplayPosition

public void **setUIDisplayPosition** (*Long uiDisplayPosition*)

setUIDisplayable

public void **setUIDisplayable** (boolean *uiDisplayable*)

setUIFilterable

public void **setUIFilterable** (boolean *uiFilterable*)

setValidations

public void **setValidations** (*List<FieldValidation> validations*)

toDto

```
public FieldDto toDto ()
```

update

```
public Field update (FieldDto field)
```

7.39.11 FieldHolder

```
public class FieldHolder
```

The main purpose of this class is to give a easy way to access values inside metadata and settings related with the given field.

Constructors**FieldHolder**

```
public FieldHolder (Field field)
```

FieldHolder

```
public FieldHolder (FieldDto field)
```

FieldHolder

```
protected FieldHolder (List<? extends Pair<String, String>> metadata, List<? extends Pair<String, ?>>  
                        settings)
```

Methods**getMetadata**

```
public String getMetadata (String name)
```

getMetadata

```
public String getMetadata (String name, String defaultValue)
```

getSetting

```
public String getSetting (String name)
```

getSetting

```
public String getSetting (String name, String defaultValue)
```

getSettingAsArray

```
public String[] getSettingAsArray (String name)
```

getSettingAsBoolean

```
public boolean getSettingAsBoolean (String name)
```

7.39.12 FieldMetadata

```
public class FieldMetadata implements Pair<String, String>
```

The FieldMetadata class contains information about a single metadata added into a field.

Constructors**FieldMetadata**

```
public FieldMetadata ()
```

FieldMetadata

```
public FieldMetadata (Field field, String key)
```

FieldMetadata

```
public FieldMetadata (MetadataDto metadata)
```

FieldMetadata

```
public FieldMetadata (Field field, String key, String value)
```

Methods**copy**

```
public FieldMetadata copy ()
```

getField

```
public Field getField ()
```

getId

public Long **getId** ()

getKey

public String **getKey** ()

getValue

public String **getValue** ()

setField

public void **setField** (Field *field*)

setId

public void **setId** (Long *id*)

setKey

public void **setKey** (String *key*)

setValue

public void **setValue** (String *value*)

toDto

public MetadataDto **toDto** ()

update

public final void **update** (MetadataDto *metadata*)

7.39.13 FieldSetting

public class **FieldSetting** implements Pair<String, String>

Constructors

FieldSetting

public **FieldSetting** ()

FieldSetting

```
public FieldSetting (Field field, TypeSetting details)
```

FieldSetting

```
public FieldSetting (Field field, TypeSetting details, String value)
```

Methods**copy**

```
public FieldSetting copy ()
```

getDetails

```
public TypeSetting getDetails ()
```

getField

```
public Field getField ()
```

getId

```
public Long getId ()
```

getKey

```
public String getKey ()
```

getValue

```
public String getValue ()
```

setDetails

```
public void setDetails (TypeSetting details)
```

setField

```
public void setField (Field field)
```

setId

```
public void setId (Long id)
```

setValue

public void **setValue** (*String value*)

toDto

public *SettingDto* **toDto** ()

7.39.14 FieldValidation

public class **FieldValidation**

The `FieldValidation` class contains the value that is related with the correct type validation and information about that whether the given validation is enabled or not.

Constructors

FieldValidation

public **FieldValidation** ()

FieldValidation

public **FieldValidation** (*Field field*, *TypeValidation details*)

FieldValidation

public **FieldValidation** (*Field field*, *TypeValidation details*, *String value*, boolean *enabled*)

Methods

copy

public *FieldValidation* **copy** ()

getDetails

public *TypeValidation* **getDetails** ()

getField

public *Field* **getField** ()

getId

public *Long* **getId** ()

getValue

```
public String getValue ()
```

isEnabled

```
public boolean isEnabled ()
```

setDetails

```
public void setDetails (TypeValidation details)
```

setEnabled

```
public void setEnabled (boolean enabled)
```

setField

```
public void setField (Field field)
```

setId

```
public void setId (Long id)
```

setValue

```
public void setValue (String value)
```

toDto

```
public ValidationCriterionDto toDto ()
```

7.39.15 Lookup

```
public class Lookup
```

The `Lookup` class contains information about single lookup

Constructors**Lookup**

```
public Lookup ()
```

Lookup

```
public Lookup (String lookupName, boolean singleObjectReturn, boolean exposedViaRest, List<Field> fields)
```

Lookup

```
public Lookup (String lookupName, boolean singleObjectReturn, boolean exposedViaRest, List<Field> fields,  
               boolean readOnly, String methodName)
```

Lookup

```
public Lookup (String lookupName, boolean singleObjectReturn, boolean exposedViaRest, List<Field> fields,  
               boolean readOnly, String methodName, List<String> rangeLookupFields, List<String> set-  
               LookupFields, Map<String, String> customOperators)
```

Lookup

```
public Lookup (String lookupName, boolean singleObjectReturn, boolean exposedViaRest, List<Field> fields,  
               boolean readOnly, String methodName, List<String> rangeLookupFields, List<String> set-  
               LookupFields, Map<String, String> customOperators, Map<String, Boolean> useGeneric-  
               Params)
```

Lookup

```
public Lookup (String lookupName, boolean singleObjectReturn, boolean exposedViaRest, List<Field> fields,  
               Entity entity)
```

Lookup

```
public Lookup (LookupDto lookupDto, List<Field> lookupFields)
```

Methods

copy

```
public Lookup copy (List<Field> fields)
```

getCustomOperators

```
public Map<String, String> getCustomOperators ()
```

getEntity

```
public Entity getEntity ()
```


getFields

```
public List<Field> getFields ()
```

getId

```
public Long getId ()
```

getLookupFieldById

```
public final Field getLookupFieldById (Long id)
```

getLookupFieldByName

```
public final Field getLookupFieldByName (String name)
```

getLookupName

```
public String getLookupName ()
```

getMethodName

```
public String getMethodName ()
```

getRangeLookupFields

```
public final List<String> getRangeLookupFields ()
```

getSetLookupFields

```
public final List<String> getSetLookupFields ()
```

getUseGenericParams

```
public Map<String, Boolean> getUseGenericParams ()
```

isExposedViaRest

```
public boolean isExposedViaRest ()
```

isRangeParam

```
public boolean isRangeParam (Field field)
```

isReadOnly

public boolean **isReadOnly** ()

isSetParam

public boolean **isSetParam** ([Field](#) *field*)

isSingleObjectReturn

public boolean **isSingleObjectReturn** ()

setCustomOperators

public void **setCustomOperators** ([Map](#)<[String](#), [String](#)> *customOperators*)

setEntity

public void **setEntity** ([Entity](#) *entity*)

setExposedViaRest

public void **setExposedViaRest** (boolean *exposedViaRest*)

setFields

public void **setFields** ([List](#)<[Field](#)> *fields*)

setId

public void **setId** ([Long](#) *id*)

setLookupName

public final void **setLookupName** ([String](#) *lookupName*)

setMethodName

public void **setMethodName** ([String](#) *methodName*)

setRangeLookupFields

public void **setRangeLookupFields** ([List](#)<[String](#)> *rangeLookupFields*)

setReadOnly

```
public void setReadOnly (boolean readOnly)
```

setSetLookupFields

```
public void setSetLookupFields (List<String> setLookupFields)
```

setSingleObjectReturn

```
public void setSingleObjectReturn (boolean singleObjectReturn)
```

setUseGenericParams

```
public void setUseGenericParams (Map<String, Boolean> useGenericParams)
```

toDto

```
public LookupDto toDto ()
```

update

```
public final void update (LookupDto lookupDto, List<Field> lookupFields)
```

7.39.16 OneToManyRelationship

```
public class OneToManyRelationship extends Relationship
```

A specialization of the `Relationship` class. Represents a one-to-many relationship.

Methods**getFieldType**

```
public String getFieldType (Field field, EntityType type)
```

getGenericSignature

```
public String getGenericSignature (Field field, EntityType type)
```

7.39.17 OneToOneRelationship

```
public class OneToOneRelationship extends Relationship
```

A specialization of the `Relationship` class. Represents a one-to-one relationship.

Methods

getFieldType

```
public String getFieldType (Field field, EntityType type)
```

getGenericSignature

```
public String getGenericSignature (Field field, EntityType type)
```

7.39.18 Relationship

```
public class Relationship
```

A class representing a relationship type. This class is inherited by different types of relationships. This class only represents the field type and provides some utility methods. It is not used in entities themselves.

Methods

getFieldType

```
public String getFieldType (Field field, EntityType type)
```

getGenericSignature

```
public String getGenericSignature (Field field, EntityType type)
```

getRelatedClassName

```
protected String getRelatedClassName (Field field, EntityType type)
```

7.39.19 RelationshipHolder

```
public class RelationshipHolder extends FieldHolder
```

The main purpose of this class is to find out how cascade should be used for the given field with relationship type.

Constructors

RelationshipHolder

```
public RelationshipHolder (Field field)
```

RelationshipHolder

```
public RelationshipHolder (ClassData data, Field field)
```

Methods

`getRelatedClass`

```
public String getRelatedClass ()
```

`isCascadeDelete`

```
public boolean isCascadeDelete ()
```

`isCascadePersist`

```
public boolean isCascadePersist ()
```

`isCascadeUpdate`

```
public boolean isCascadeUpdate ()
```

`isOneToMany`

```
public boolean isOneToMany ()
```

`isOneToOne`

```
public boolean isOneToOne ()
```

7.39.20 RestOptions

```
public class RestOptions
```

The `RestOptions` class representing rest options of given entity. This class is related with table in database with the same name.

Constructors

`RestOptions`

```
public RestOptions ()
```

`RestOptions`

```
public RestOptions (Entity entity)
```

Methods

copy

public [RestOptions](#) **copy** ()

getEntity

public [Entity](#) **getEntity** ()

getFields

public [List](#)<[Field](#)> **getFields** ()

getId

public [Long](#) **getId** ()

getLookups

public [List](#)<[Lookup](#)> **getLookups** ()

isAllowCreate

public boolean **isAllowCreate** ()

isAllowDelete

public boolean **isAllowDelete** ()

isAllowRead

public boolean **isAllowRead** ()

isAllowUpdate

public boolean **isAllowUpdate** ()

setAllowCreate

public void **setAllowCreate** (boolean *allowCreate*)

setAllowDelete

public void **setAllowDelete** (boolean *allowDelete*)

setAllowRead

```
public void setAllowRead (boolean allowRead)
```

setAllowUpdate

```
public void setAllowUpdate (boolean allowUpdate)
```

setEntity

```
public void setEntity (Entity entity)
```

setId

```
public void setId (Long id)
```

toDto

```
public RestOptionsDto toDto ()
```

update

```
public final void update (RestOptionsDto restOptionsDto)
```

7.39.21 Tracking

```
public class Tracking
```

The `Tracking` contains information about which fields and what kind of actions should be logged. This class is related with table in database with the same name.

Constructors**Tracking**

```
public Tracking ()
```

Tracking

```
public Tracking (Entity entity)
```

Methods**copy**

```
public Tracking copy ()
```

getEntity

public [Entity](#) **getEntity** ()

getFields

public [List](#)<[Field](#)> **getFields** ()

getId

public [Long](#) **getId** ()

isAllowCreate

public boolean **isAllowCreate** ()

isAllowDelete

public boolean **isAllowDelete** ()

isAllowRead

public boolean **isAllowRead** ()

isAllowUpdate

public boolean **isAllowUpdate** ()

setAllowCreate

public void **setAllowCreate** (boolean *allowCreate*)

setAllowDelete

public void **setAllowDelete** (boolean *allowDelete*)

setAllowRead

public void **setAllowRead** (boolean *allowRead*)

setAllowUpdate

public void **setAllowUpdate** (boolean *allowUpdate*)

setEntity

```
public void setEntity (Entity entity)
```

setId

```
public void setId (Long id)
```

toDto

```
public TrackingDto toDto ()
```

7.39.22 Type

```
public class Type
```

The `Type` class contains information about a single type in mds system. The mds type can have a settings and validations that can be assigned to field with the same type.

Constructors

Type

```
public Type ()
```

Type

```
public Type (Class<?> typeClass)
```

Type

```
public Type (String displayName, String description, Class<?> typeClass)
```

Methods

getDefaultName

```
public String getDefaultName ()
```

getDescription

```
public String getDescription ()
```

getDisplayName

```
public String getDisplayName ()
```

getId

public Long **getId** ()

getSettings

public List<TypeSetting> **getSettings** ()

getTypeClass

public Class<?> **getTypeClass** ()

getTypeClassName

public String **getTypeClassName** ()

getValidations

public List<TypeValidation> **getValidations** ()

hasSettings

public boolean **hasSettings** ()

hasValidation

public boolean **hasValidation** ()

isBlob

public boolean **isBlob** ()

isCombobox

public boolean **isCombobox** ()

isRelationship

public boolean **isRelationship** ()

parse

public Object **parse** (String *str*)

setDefaultName

```
public void setDefaultName (String defaultName)
```

setDescription

```
public void setDescription (String description)
```

setDisplayName

```
public void setDisplayName (String displayName)
```

setId

```
public void setId (Long id)
```

setSettings

```
public void setSettings (List<TypeSetting> settings)
```

setTypeClass

```
public void setTypeClass (Class<?> typeClass)
```

setValidations

```
public void setValidations (List<TypeValidation> validations)
```

toDto

```
public TypeDto toDto ()
```

7.39.23 TypeSetting

```
public class TypeSetting
```

The `TypeSetting` contains settings for the given mds type. This class is related with table in database with the same name.

Constructors**TypeSetting**

```
public TypeSetting ()
```

TypeSetting

public **TypeSetting** (*String name*)

Methods

getDefaultValue

public *String* **getDefaultValue** ()

getId

public *Long* **getId** ()

getName

public *String* **getName** ()

getTypeSettingOptions

public *List*<*TypeSettingOption*> **getTypeSettingOptions** ()

getValueType

public *Type* **getValueType** ()

setDefaultValue

public void **setDefaultValue** (*String defaultValue*)

setId

public void **setId** (*Long id*)

setName

public void **setName** (*String name*)

setTypeSettingOptions

public void **setTypeSettingOptions** (*List*<*TypeSettingOption*> *typeSettingOptions*)

setValueType

public void **setValueType** (*Type valueType*)

7.39.24 TypeSettingOption

public class **TypeSettingOption**

The `TypeSettingOption` contains a single setting option for the given type setting. This class is related with table in database with the same name.

Constructors

TypeSettingOption

public **TypeSettingOption** (*String name*)

Methods

getId

public *Long* **getId** ()

getName

public *String* **getName** ()

setId

public void **setId** (*Long id*)

setName

public void **setName** (*String name*)

7.39.25 TypeValidation

public class **TypeValidation**

The `TypeValidation` contains a single validation option for the given type. This class is related with table in database with the same name.

Constructors

TypeValidation

public **TypeValidation** ()

TypeValidation

public **TypeValidation** (*String displayName*, *Type valueType*)

Methods

getAnnotations

```
public List<Class<? extends Annotation>> getAnnotations ()
```

getDisplayName

```
public String getDisplayName ()
```

getId

```
public Long getId ()
```

getValueType

```
public Type getValueType ()
```

setAnnotations

```
public void setAnnotations (List<Class<? extends Annotation>> annotations)
```

setDisplayName

```
public void setDisplayName (String displayName)
```

setId

```
public void setId (Long id)
```

setValueType

```
public void setValueType (Type valueType)
```

toString

```
public String toString ()
```

7.40 org.motechproject.mds.dto

7.40.1 AdvancedSettingsDto

```
public class AdvancedSettingsDto
```

The `AdvancedSettingsDto` contains information about advanced settings of an entity.

Methods

addNewIndex

```
public void addNewIndex (String lookupName)
```

equals

```
public boolean equals (Object obj)  
    { @inheritDoc }
```

getBrowsing

```
public BrowsingSettingsDto getBrowsing ()
```

getEntityId

```
public Long getEntityId ()
```

getId

```
public Long getId ()
```

getIndexes

```
public List<LookupDto> getIndexes ()
```

getRestOptions

```
public RestOptionsDto getRestOptions ()
```

getTracking

```
public TrackingDto getTracking ()
```

hashCode

```
public int hashCode ()  
    { @inheritDoc }
```

removeIndex

```
public void removeIndex (Integer idx)
```

setBrowsing

public void **setBrowsing** ([BrowsingSettingsDto](#) *browsing*)

setEntityId

public void **setEntityId** ([Long](#) *entityId*)

setId

public void **setId** ([Long](#) *id*)

setIndexes

public void **setIndexes** ([List](#)<[LookupDto](#)> *indexes*)

setRestOptions

public void **setRestOptions** ([RestOptionsDto](#) *restOptions*)

setTracking

public void **setTracking** ([TrackingDto](#) *tracking*)

toString

```
public String toString ()  
    { @inheritDoc }
```

7.40.2 BrowsingSettingsDto

public class **BrowsingSettingsDto**

The [BrowsingSettingsDto](#) contains informations about filed browsing settings

Methods

addDisplayedField

public void **addDisplayedField** ([Number](#) *id*)

addFilterableField

public void **addFilterableField** ([Number](#) *id*)

containsDisplayedField

```
public boolean containsDisplayedField (Long number)
```

containsFilterableField

```
public boolean containsFilterableField (Number id)
```

equals

```
public boolean equals (Object obj)  
    { @inheritDoc }
```

getDisplayedFields

```
public List<Number> getDisplayedFields ()
```

getFilterableFields

```
public List<Number> getFilterableFields ()
```

hashCode

```
public int hashCode ()  
    { @inheritDoc }
```

indexOfDisplayedField

```
public long indexOfDisplayedField (Long id)
```

removeFilterableField

```
public void removeFilterableField (Number id)
```

setDisplayedFields

```
public void setDisplayedFields (List<Number> displayedFields)
```

setFilterableFields

```
public void setFilterableFields (List<Number> filterableFields)
```

toString

```
public String toString()  
    { @inheritDoc }
```

7.40.3 DraftData

```
public class DraftData
```

The `DraftData` contains information that are used later for creating temporary changes in field.

Fields**ADD_NEW_INDEX**

```
public static final String ADD_NEW_INDEX
```

ADVANCED

```
public static final String ADVANCED
```

DISPLAY_NAME

```
public static final String DISPLAY_NAME
```

FIELD

```
public static final String FIELD
```

FIELD_ID

```
public static final String FIELD_ID
```

NAME

```
public static final String NAME
```

PATH

```
public static final String PATH
```

REMOVE_INDEX

```
public static final String REMOVE_INDEX
```

SECURITY

public static final [String](#) **SECURITY**

TYPE_CLASS

public static final [String](#) **TYPE_CLASS**

VALUE

public static final [String](#) **VALUE**

Methods

getPath

public [String](#) **getPath** ()

getValue

public [Object](#) **getValue** ([String](#) *key*)

getValues

public [Map](#)<[String](#), [Object](#)> **getValues** ()

isCreate

public boolean **isCreate** ()

isEdit

public boolean **isEdit** ()

isForAdvanced

public boolean **isForAdvanced** ()

isForField

public boolean **isForField** ()

isForSecurity

public boolean **isForSecurity** ()

isRemove

public boolean **isRemove** ()

setCreate

public void **setCreate** (boolean *create*)

setEdit

public void **setEdit** (boolean *edit*)

setRemove

public void **setRemove** (boolean *remove*)

setValues

public void **setValues** (Map<String, Object> *values*)

7.40.4 DraftResult

public class **DraftResult** implements [Serializable](#)

After users do draft changes an instance of this class is returned. It contains information about the draft state.

Constructors

DraftResult

public **DraftResult** (boolean *changesMade*, boolean *outdated*)

Methods

isChangesMade

public boolean **isChangesMade** ()

isOutdated

public boolean **isOutdated** ()

setChangesMade

public void **setChangesMade** (boolean *changesMade*)

setOutdated

public void **setOutdated** (boolean *outdated*)

7.40.5 EntityDto

public class **EntityDto**

The `EntityDto` class contains only basic information about an entity like id, name, module and namespace.

Constructors

EntityDto

public **EntityDto** ()

EntityDto

public **EntityDto** (String *className*)

EntityDto

public **EntityDto** (Long *id*, String *className*)

EntityDto

public **EntityDto** (String *className*, SecurityMode *securityMode*, Set<String> *securityMembers*)

EntityDto

public **EntityDto** (Long *id*, String *className*, SecurityMode *securityMode*, Set<String> *securityMembers*)

EntityDto

public **EntityDto** (Long *id*, String *className*, String *module*, SecurityMode *securityMode*, Set<String> *securityMembers*)

EntityDto

public **EntityDto** (Long *id*, String *className*, String *module*, String *namespace*, SecurityMode *securityMode*, Set<String> *securityMembers*)

EntityDto

public **EntityDto** (String *className*, String *name*, String *module*, String *namespace*, SecurityMode *securityMode*, Set<String> *securityMembers*)

EntityDto

```
public EntityDto (Long id, String className, String name, String module, String namespace, SecurityMode
                 securityMode, Set<String> securityMembers)
```

EntityDto

```
public EntityDto (Long id, String className, String name, String module, String namespace, SecurityMode
                 securityMode, Set<String> securityMembers, String superClass)
```

EntityDto

```
public EntityDto (Long id, String className, String name, String module, String namespace, SecurityMode
                 securityMode, Set<String> securityMembers, String superClass, boolean abstractClass)
```

Methods

equals

```
public boolean equals (Object obj)
    { @inheritDoc }
```

getClassName

```
public String getClassName ()
```

getId

```
public Long getId ()
```

getModule

```
public String getModule ()
```

getName

```
public String getName ()
```

getNamespace

```
public String getNamespace ()
```

getSecurityMembers

```
public Set<String> getSecurityMembers ()
```

getSecurityMode

```
public SecurityMode getSecurityMode ()
```

getSuperClass

```
public String getSuperClass ()
```

hashCode

```
public int hashCode ()  
    { @inheritDoc }
```

isAbstractClass

```
public boolean isAbstractClass ()
```

isDDE

```
public boolean isDDE ()
```

isModified

```
public boolean isModified ()
```

isOutdated

```
public boolean isOutdated ()
```

isReadOnly

```
public boolean isReadOnly ()
```

setAbstractClass

```
public void setAbstractClass (boolean abstractClass)
```

setClassName

```
public void setClassName (String className)
```

setId

```
public void setId (Long id)
```

setModified

public void **setModified** (boolean *modified*)

setModule

public void **setModule** ([String](#) *module*)

setName

public void **setName** ([String](#) *name*)

setNamespace

public void **setNamespace** ([String](#) *namespace*)

setOutdated

public void **setOutdated** (boolean *outdated*)

setReadOnly

public void **setReadOnly** (boolean *readOnly*)

setSecurityMembers

public void **setSecurityMembers** ([Set](#)<[String](#)> *securityMembers*)

setSecurityMode

public void **setSecurityMode** ([SecurityMode](#) *securityMode*)

setSuperClass

public void **setSuperClass** ([String](#) *superClass*)

toString

```
public String toString ()  
    { @inheritDoc }
```

7.40.6 FieldBasicDto

public class **FieldBasicDto**

The `FieldBasicDto` contains basic information about a field.

Constructors

FieldBasicDto

```
public FieldBasicDto ()
```

FieldBasicDto

```
public FieldBasicDto (String displayName, String name)
```

FieldBasicDto

```
public FieldBasicDto (String displayName, String name, boolean required, Object defaultValue, String tooltip)
```

Methods

equals

```
public boolean equals (Object obj)  
    { @inheritDoc }
```

getDefaultValue

```
public Object getDefaultValue ()
```

getDisplayName

```
public String getDisplayName ()
```

getName

```
public String getName ()
```

getTooltip

```
public String getTooltip ()
```

hashCode

```
public int hashCode ()  
    { @inheritDoc }
```

isRequired

```
public boolean isRequired()
```

setDefaultValue

```
public void setDefaultValue(Object defaultValue)
```

setDisplayName

```
public void setDisplayName(String displayName)
```

setName

```
public void setName(String name)
```

setRequired

```
public void setRequired(boolean required)
```

setTooltip

```
public void setTooltip(String tooltip)
```

toString

```
public String toString()  
    { @inheritDoc }
```

7.40.7 FieldDto

```
public class FieldDto
```

The `FieldDto` class contains information about an existing field in an entity.

Constructors**FieldDto**

```
public FieldDto()
```

FieldDto

```
public FieldDto(Long id, Long entityId, TypeDto type, FieldBasicDto basic, boolean readOnly,  
                List<MetadataDto> metadata, FieldValidationDto validation, List<SettingDto> settings,  
                List<LookupDto> lookups)
```

FieldDto

```
public FieldDto (Long id, Long entityId, TypeDto type, FieldBasicDto basic, boolean readOnly, FieldValidationDto validation)
```

Methods

addEmptyMetadata

```
public void addEmptyMetadata ()
```

addMetadata

```
public void addMetadata (MetadataDto metadata)
```

equals

```
public boolean equals (Object obj)  
    { @inheritDoc }
```

getBasic

```
public FieldBasicDto getBasic ()
```

getEntityId

```
public Long getEntityId ()
```

getId

```
public Long getId ()
```

getLookups

```
public List<LookupDto> getLookups ()
```

getMetadata

```
public List<MetadataDto> getMetadata ()
```

getMetadata

```
public MetadataDto getMetadata (String key)
```

getSetting

public [SettingDto](#) **getSetting** ([String](#) *name*)

getSettings

public [List](#)<[SettingDto](#)> **getSettings** ()

getType

public [TypeDto](#) **getType** ()

getValidation

public [FieldValidationDto](#) **getValidation** ()

hashCode

```
public int hashCode ()  
    { @inheritDoc }
```

isReadOnly

public boolean **isReadOnly** ()

removeMetadata

public void **removeMetadata** ([Integer](#) *idx*)

setBasic

public void **setBasic** ([FieldBasicDto](#) *basic*)

setEntityId

public void **setEntityId** ([Long](#) *entityId*)

setId

public void **setId** ([Long](#) *id*)

setLookups

public void **setLookups** ([List](#)<[LookupDto](#)> *lookups*)

setMetadata

```
public void setMetadata (List<MetadataDto> metadata)
```

setReadOnly

```
public void setReadOnly (boolean readOnly)
```

setSettings

```
public void setSettings (List<SettingDto> settings)
```

setType

```
public void setType (TypeDto type)
```

setValidation

```
public void setValidation (FieldValidationDto validation)
```

toString

```
public String toString ()  
    { @inheritDoc }
```

7.40.8 FieldInstanceDto

```
public class FieldInstanceDto
```

The `FieldInstanceDto` class contains information about an existing field in an instance.

Constructors

FieldInstanceDto

```
public FieldInstanceDto ()
```

FieldInstanceDto

```
public FieldInstanceDto (Long id, Long instanceId, FieldBasicDto basic)
```

Methods

equals

```
public boolean equals (Object obj)  
    { @inheritDoc }
```

getBasic

```
public FieldBasicDto getBasic ()
```

getId

```
public Long getId ()
```

getInstanceId

```
public Long getInstanceId ()
```

hashCode

```
public int hashCode ()  
    { @inheritDoc }
```

setBasic

```
public void setBasic (FieldBasicDto basic)
```

setId

```
public void setId (Long id)
```

setInstanceId

```
public void setInstanceId (Long instanceId)
```

toString

```
public String toString ()  
    { @inheritDoc }
```

7.40.9 FieldValidationDto

```
public class FieldValidationDto
```

The `FieldValidationDto` class contains information about validation criteria for field.

Fields

DOUBLE

```
public static final FieldValidationDto DOUBLE
```

Constant `DOUBLE` contains validation criteria for double type.

INTEGER

public static final [FieldValidationDto](#) **INTEGER**

Constant **INTEGER** contains validation criteria for integer type.

STRING

public static final [FieldValidationDto](#) **STRING**

Constant **STRING** contains validation criteria for string type.

Constructors

FieldValidationDto

public **FieldValidationDto** ()

FieldValidationDto

public **FieldValidationDto** ([ValidationCriterionDto](#)... *criteria*)

Methods

addCriterion

public void **addCriterion** ([ValidationCriterionDto](#) *criterion*)

equals

public boolean **equals** ([Object](#) *obj*)
{ @inheritDoc }

getCriteria

public [List](#)<[ValidationCriterionDto](#)> **getCriteria** ()

getCriterion

public [ValidationCriterionDto](#) **getCriterion** ([String](#) *displayName*)

hashCode

public int **hashCode** ()
{ @inheritDoc }

setCriteria

```
public void setCriteria (List<ValidationCriterionDto> criteria)
```

toString

```
public String toString ()  
    { @inheritDoc }
```

7.40.10 LookupDto

```
public class LookupDto
```

The `LookupDto` class contains information about single lookup defined by user

Constructors**LookupDto**

```
public LookupDto ()
```

LookupDto

```
public LookupDto (String lookupName, boolean singleObjectReturn, boolean exposedViaRest)
```

LookupDto

```
public LookupDto (String lookupName, boolean singleObjectReturn, boolean exposedViaRest,  
    List<LookupFieldDto> lookupFields, boolean readOnly)
```

LookupDto

```
public LookupDto (String lookupName, boolean singleObjectReturn, boolean exposedViaRest,  
    List<LookupFieldDto> lookupFields, boolean readOnly, String methodName)
```

LookupDto

```
public LookupDto (Long id, String lookupName, boolean singleObjectReturn, boolean exposedViaRest,  
    List<LookupFieldDto> lookupFields, boolean readOnly, String methodName)
```

Methods**addField**

```
public void addField (Long field)
```


addField

```
public void addField (Integer field)
```

equals

```
public boolean equals (Object o)  
    { @inheritDoc }
```

getId

```
public Long getId ()
```

getLookupFields

```
public final List<LookupFieldDto> getLookupFields ()
```

getLookupName

```
public String getLookupName ()
```

getMethodName

```
public String getMethodName ()
```

hashCode

```
public int hashCode ()  
    { @inheritDoc }
```

insertField

```
public void insertField (Integer idx, Integer fieldId)
```

insertField

```
public void insertField (Integer idx, Long fieldId)
```

isExposedViaRest

```
public boolean isExposedViaRest ()
```

isReadOnly

```
public boolean isReadOnly ()
```

isSingleObjectReturn

```
public boolean isSingleObjectReturn ()
```

removeField

```
public void removeField (Long fieldId)
```

removeField

```
public void removeField (Integer fieldId)
```

setExposedViaRest

```
public void setExposedViaRest (boolean isExposedViaRest)
```

setId

```
public void setId (Long id)
```

setLookupFields

```
public void setLookupFields (List<LookupFieldDto> lookupFields)
```

setLookupName

```
public void setLookupName (String lookupName)
```

setMethodName

```
public void setMethodName (String methodName)
```

setReadOnly

```
public void setReadOnly (boolean readOnly)
```

setSingleObjectReturn

```
public void setSingleObjectReturn (boolean singleObjectReturn)
```

toString

```
public String toString ()  
    { @inheritDoc }
```

7.40.11 LookupFieldDto

public class **LookupFieldDto**

Represents a field added to a lookup. The lookup using a given field can be done using multiple lookup types.

Constructors

LookupFieldDto

public **LookupFieldDto** ()

LookupFieldDto

public **LookupFieldDto** (Long *id*, String *name*, Type *type*)

LookupFieldDto

public **LookupFieldDto** (Long *id*, String *name*, Type *type*, String *customOperator*)

LookupFieldDto

public **LookupFieldDto** (Long *id*, String *name*, Type *type*, String *customOperator*, boolean *useGenericParam*)

Methods

equals

public boolean **equals** (Object *o*)

getCustomOperator

public String **getCustomOperator** ()

getId

public Long **getId** ()

getName

public String **getName** ()

getType

public Type **getType** ()

hashCode

public int **hashCode** ()

isUseGenericParam

public boolean **isUseGenericParam** ()

setCustomOperator

public void **setCustomOperator** (*String customOperator*)

setId

public void **setId** (*Long id*)

setName

public void **setName** (*String name*)

setType

public void **setType** (*Type type*)

setUseGenericParam

public void **setUseGenericParam** (boolean *useGenericParam*)

7.40.12 LookupFieldDto.Type

public static enum **Type**

The lookup type represents whether the lookup will be done by comparing to a single field, matching values to a range, or matching to a set of values.

Enum Constants

RANGE

public static final *LookupFieldDto.Type* **RANGE**

SET

public static final *LookupFieldDto.Type* **SET**

VALUE

public static final [LookupFieldDto.Type](#) **VALUE**

7.40.13 MetadataDto

public class **MetadataDto** implements [Pair<String, String>](#)

The `MetadataDto` contains key and value of a single field metadata.

Constructors

MetadataDto

public **MetadataDto** ()

MetadataDto

public **MetadataDto** ([String](#) *key*, [String](#) *value*)

MetadataDto

public **MetadataDto** ([Long](#) *id*, [String](#) *key*, [String](#) *value*)

Methods

equals

public boolean **equals** ([Object](#) *obj*)
 { [@inheritDoc](#) }

getId

public [Long](#) **getId** ()

getKey

public [String](#) **getKey** ()

getValue

public [String](#) **getValue** ()

hashCode

```
public int hashCode ()  
    { @inheritDoc }
```

setId

```
public void setId (Long id)
```

setKey

```
public void setKey (String key)
```

setValue

```
public void setValue (String value)
```

toString

```
public String toString ()  
    { @inheritDoc }
```

7.40.14 RestOptionsDto

```
public class RestOptionsDto  
    Class representing rest options of given entity.
```

Constructors**RestOptionsDto**

```
public RestOptionsDto ()
```

RestOptionsDto

```
public RestOptionsDto (boolean create, boolean read, boolean update, boolean delete)
```

Methods**addField**

```
public void addField (Number id)
```

addLookup

```
public void addLookup (Number id)
```

containsFieldId

```
public boolean containsFieldId (Number id)
```

containsLookupId

```
public boolean containsLookupId (Number id)
```

equals

```
public boolean equals (Object obj)  
    { @inheritDoc }
```

getFieldIds

```
public List<Number> getFieldIds ()
```

getId

```
public Long getId ()
```

getLookupIds

```
public List<Number> getLookupIds ()
```

hashCode

```
public int hashCode ()  
    { @inheritDoc }
```

isCreate

```
public boolean isCreate ()
```

isDelete

```
public boolean isDelete ()
```

isRead

```
public boolean isRead ()
```

isUpdate

public boolean **isUpdate** ()

removeField

public void **removeField** (Number *id*)

removeLookup

public void **removeLookup** (Number *id*)

setCreate

public void **setCreate** (boolean *create*)

setDelete

public void **setDelete** (boolean *delete*)

setFieldIds

public void **setFieldIds** (List<Number> *fieldIds*)

setId

public void **setId** (Long *id*)

setLookupIds

public void **setLookupIds** (List<Number> *lookupIds*)

setRead

public void **setRead** (boolean *read*)

setUpdate

public void **setUpdate** (boolean *update*)

toString

```
public String toString ()  
    { @inheritDoc }
```


7.40.15 SettingDto

public class **SettingDto** implements [Pair<String, Object>](#)

The `SettingDto` contains information about a single setting inside a field.

Constructors

SettingDto

```
public SettingDto ()
```

SettingDto

```
public SettingDto (String name, Object value)
```

SettingDto

```
public SettingDto (String name, Object value, TypeDto type, SettingOptions... options)
```

Methods

copy

```
public SettingDto copy ()
```

equals

```
public boolean equals (Object obj)  
    { @inheritDoc }
```

getKey

```
public String getKey ()
```

getName

```
public String getName ()
```

getOptions

```
public List<SettingOptions> getOptions ()
```

getType

```
public TypeDto getType ()
```

getValue

```
public Object getValue ()
```

getValueAsString

```
public String getValueAsString ()
```

hashCode

```
public int hashCode ()  
    { @inheritDoc }
```

setName

```
public void setName (String name)
```

setOptions

```
public void setOptions (List<SettingOptions> options)
```

setType

```
public void setType (TypeDto type)
```

setValue

```
public void setValue (Object value)
```

toString

```
public String toString ()  
    { @inheritDoc }
```

7.40.16 SettingOptions

```
public enum SettingOptions
```

The `SettingOptions` contains available options that can be added to field setting.

Enum Constants**POSITIVE**

```
public static final SettingOptions POSITIVE
```

Ensure that a value in a given setting is a number and it has a positive value.

REQUIRE

public static final [SettingOptions](#) **REQUIRE**
Force setting a value for a given setting.

7.40.17 TrackingDto

public class **TrackingDto**
The `TrackingDto` contains information about which fields and what kind of actions should be logged.

Methods

addAction

public void **addAction** ([String](#) *action*)

addField

public void **addField** ([Number](#) *fieldId*)

equals

public boolean **equals** ([Object](#) *obj*)
{ @inheritDoc }

getActions

public [List](#)<[String](#)> **getActions** ()

getFields

public [List](#)<[Long](#)> **getFields** ()

hashCode

public int **hashCode** ()
{ @inheritDoc }

isAllowCreate

public boolean **isAllowCreate** ()

isAllowDelete

public boolean **isAllowDelete** ()

isAllowRead

```
public boolean isAllowRead ()
```

isAllowUpdate

```
public boolean isAllowUpdate ()
```

removeAction

```
public void removeAction (String action)
```

removeField

```
public void removeField (Number fieldId)
```

setActions

```
public void setActions (List<String> actions)
```

setFields

```
public void setFields (List<Long> fields)
```

toString

```
public String toString ()  
    { @inheritDoc }
```

7.40.18 TypeDto

```
public class TypeDto
```

The `TypeDto` class contains information about an available field in an entity.

Fields

BLOB

```
public static final TypeDto BLOB
```

Constant `BLOB` is a representation of the MDS BLOB type.

BOOLEAN

```
public static final TypeDto BOOLEAN
```

Constant `BOOLEAN` is a representation of the MDS Boolean type.

DATE

public static final [TypeDto](#) **DATE**

Constant **DATE** is a representation of the MDS Date type.

DATETIME

public static final [TypeDto](#) **DATETIME**

Constant **DATETIME** is a representation of the MDS DateTime type.

DOUBLE

public static final [TypeDto](#) **DOUBLE**

Constant **DOUBLE** is a representation of the MDS Decimal type.

INTEGER

public static final [TypeDto](#) **INTEGER**

Constant **INTEGER** is a representation of the MDS Integer type.

LIST

public static final [TypeDto](#) **LIST**

Constant **LIST** is a representation of the MDS Combobox type.

LOCAL_DATE

public static final [TypeDto](#) **LOCAL_DATE**

Constant **LOCAL_DATE** is a representation of the `org.joda.time.LocalDate` type.

LONG

public static final [TypeDto](#) **LONG**

Constant **LONG** is a representation of the MDS Long type.

MAP

public static final [TypeDto](#) **MAP**

Constant **MAP** is a representation of the MDS Map type.

PERIOD

public static final [TypeDto](#) **PERIOD**

Constant **PERIOD** is a representation of the MDS Period type.

STRING

public static final [TypeDto](#) **STRING**

Constant `STRING` is a representation of the MDS String type.

TIME

public static final [TypeDto](#) **TIME**

Constant `TIME` is a representation of the MDS Time type.

Constructors

TypeDto

public **TypeDto** ()

TypeDto

public **TypeDto** (*String displayName*, *String description*, *String defaultName*, *String typeClass*)

TypeDto

public **TypeDto** (*Long id*, *String displayName*, *String description*, *String defaultName*, *String typeClass*)

Methods

equals

public boolean **equals** (*Object obj*)
 { @inheritDoc }

getDefaultName

public [String](#) **getDefaultName** ()

getDescription

public [String](#) **getDescription** ()

getDisplayName

public [String](#) **getDisplayName** ()

getId

```
public Long getId ()
```

getTypeClass

```
public String getTypeClass ()
```

hashCode

```
public int hashCode ()  
    { @inheritDoc }
```

isCombobox

```
public boolean isCombobox ()
```

isRelationship

```
public boolean isRelationship ()
```

setDefaultName

```
public void setDefaultName (String defaultName)
```

setDescription

```
public void setDescription (String description)
```

setDisplayName

```
public void setDisplayName (String displayName)
```

setId

```
public void setId (Long id)
```

setTypeClass

```
public void setTypeClass (String typeClass)
```

toString

```
public String toString ()  
    { @inheritDoc }
```

7.40.19 ValidationCriterionDto

public class **ValidationCriterionDto**

The `ValidationCriterionDto` contains information about single criterion for field validation.

Constructors

ValidationCriterionDto

public **ValidationCriterionDto** ()

ValidationCriterionDto

public **ValidationCriterionDto** (*String displayName*, *TypeDto type*)

ValidationCriterionDto

public **ValidationCriterionDto** (*String displayName*, *TypeDto type*, *Object value*, boolean *enabled*)

Methods

equals

public boolean **equals** (*Object obj*)
 { @inheritDoc }

getDisplayName

public *String* **getDisplayName** ()

getType

public *TypeDto* **getType** ()

getValue

public *Object* **getValue** ()

hashCode

public int **hashCode** ()
 { @inheritDoc }

isEnabled

public boolean **isEnabled** ()

setDisplayNames

```
public void setDisplayNames (String displayName)
```

setEnabled

```
public void setEnabled (boolean enabled)
```

setType

```
public void setType (TypeDto type)
```

setValue

```
public void setValue (Object value)
```

toString

```
public String toString ()  
    { @inheritDoc }
```

valueAsString

```
public String valueAsString ()
```

7.41 org.motechproject.mds.enhancer

7.41.1 MdsJDOEnhancer

public class **MdsJDOEnhancer** extends JDOEnhancer

The `MdsJDOEnhancer` class is a wrapper for `org.datanucleus.api.jdo.JDOEnhancer` class. Its task is to add the missing information into created entity class.

Constructors

MdsJDOEnhancer

```
public MdsJDOEnhancer (Properties config, ClassLoader classLoader)
```

Methods

addClass

```
public void addClass (ClassData classData)
```

7.42 org.motechproject.mds.ex

7.42.1 EmptyTrashException

public class **EmptyTrashException** extends [MdsException](#)

The `EmptyTrashException` exception signals a situation that there were some problems with cleaning the module trash.

Constructors

EmptyTrashException

public **EmptyTrashException** ([Throwable](#) *cause*)

Constructs a new `EmptyTrashException` with `mds.error.emptyTrashException` as a message key.

Parameters

- **cause** – the cause of exception.

7.42.2 EntityAlreadyExistException

public class **EntityAlreadyExistException** extends [MdsException](#)

The `EntityAlreadyExistException` exception signals a situation in which a user wants to create a new entity with a name that already exist in database.

Constructors

EntityAlreadyExistException

public **EntityAlreadyExistException** ()

Constructs a new `EntityAlreadyExistException` with `mds.error.entityAlreadyExist` as a message key.

7.42.3 EntityChangedException

public class **EntityChangedException** extends [MdsException](#)

This exception signals that an Entity was changed(presumably by another user).

Constructors

EntityChangedException

public **EntityChangedException** ()

7.42.4 EntityCreationException

public class **EntityCreationException** extends [MdsException](#)

The `EntityCreationException` exception signals a situation when there were problems with creating new entity class.

Constructors

EntityCreationException

public **EntityCreationException** ([Throwable](#) *cause*)

Constructs a new EntityCreationException with *mds.error.entityBuilderFailure* as a message key.

Parameters

- **cause** – the cause of exception.

EntityCreationException

public **EntityCreationException** ([String](#) *messageKey*)

7.42.5 EntityDeletedException

public class **EntityDeletedException** extends [MdsException](#)

This exception signals that the Entity was deleted(presumably by an another user).

Constructors

EntityDeletedException

public **EntityDeletedException** ()

7.42.6 EntityInfrastructureException

public class **EntityInfrastructureException** extends [MdsException](#)

The EntityInfrastructureException exception signals a situation when there were problems with creating repository/service interface/service class for entity.

Constructors

EntityInfrastructureException

public **EntityInfrastructureException** ([Throwable](#) *cause*)

Constructs a new EntityInfrastructureException with *mds.error.entityInfrastructureFailure* as a message key.

Parameters

- **cause** – the cause of exception.

7.42.7 EntityNotFoundException

public class **EntityNotFoundException** extends [MdsException](#)

The EntityNotFoundException exception signals a situation in which an entity with a given id does not exist in database.

Constructors

EntityNotFoundException

public **EntityNotFoundException** ()

Constructs a new EntityNotFoundException with *mds.error.entityNotFound* as a message key.

7.42.8 EntityReadOnlyException

public class **EntityReadOnlyException** extends [MdsException](#)

The `EntityReadOnlyException` exception signals a situation in which a user wants to make changes on an entity which is read only (it was created by a module).

Constructors

EntityReadOnlyException

public **EntityReadOnlyException** ()

Constructs a new EntityReadOnlyException with *mds.error.entityIsReadOnly* as a message key.

7.42.9 EntitySchemaMismatchException

public class **EntitySchemaMismatchException** extends [MdsException](#)

The `EntitySchemaMismatch` exception signals a situation in which a user wants to revert their instance to a version on a different schema version.

Constructors

EntitySchemaMismatchException

public **EntitySchemaMismatchException** ()

Constructs a new EntitySchemaMismatch with *mds.error.entitySchemaMismatch* as a message key.

7.42.10 FieldNotFoundException

public class **FieldNotFoundException** extends [MdsException](#)

This exception signals that a given field was not found for the Entity.

Constructors

FieldNotFoundException

public **FieldNotFoundException** ()

7.42.11 FieldUsedInLookupException

public class **FieldUsedInLookupException** extends [MdsException](#)

Exception indicating that a field cannot be removed, since it is used in a lookup.

Constructors

FieldUsedInLookupException

public **FieldUsedInLookupException** (*String fieldName*, *String lookupNames*)

7.42.12 IllegalLookupException

public class **IllegalLookupException** extends [RuntimeException](#)

Signales that the user defined an illegal lookup.

Constructors

IllegalLookupException

public **IllegalLookupException** (*String message*)

7.42.13 LoaderException

public class **LoaderException** extends [RuntimeException](#)

The `LoaderException` exception signals situations in which there were problems with correct loading the given class or its dependencies.

Constructors

LoaderException

public **LoaderException** (*Throwable cause*)

7.42.14 LookupExecutionException

public class **LookupExecutionException** extends [MdsException](#)

Signals that we were not able to execute a lookup for a given entity.

Constructors

LookupExecutionException

public **LookupExecutionException** ()

LookupExecutionException

public **LookupExecutionException** (*Throwable cause*)

7.42.15 LookupNameIsRepeatedException

public class **LookupNameIsRepeatedException** extends [MdsException](#)

The `LookupNameIsRepeatedException` exception signals a situation when are more than one lookups in the entity with the same name.

Constructors

LookupNameIsRepeatedException

public **LookupNameIsRepeatedException** ()

7.42.16 LookupNotFoundException

public class **LookupNotFoundException** extends [MdsException](#)

The `LookupNotFoundException` exception signals a situation in which a lookup with given id does not exist in database.

Constructors

LookupNotFoundException

public **LookupNotFoundException** ()

Constructs a new `LookupNotFoundException` with `mds.error.lookupNotFound` as a message key.

7.42.17 MdsException

public class **MdsException** extends [RuntimeException](#)

The `MdsException` exception is a basic class for all other exceptions defined in the mds module. It contains information about a message key which will be used on UI to present a message in appropriate language.

Constructors

MdsException

public **MdsException** (*String messageKey*)

Constructs a new mds exception with the specified message key.

Parameters

- **messageKey** – the message key used later to display message in appropriate language on UI.

MdsException

public **MdsException** (*String* messageKey, *String* params)

Constructs a new mds exception with the specified message key and params.

Parameters

- **messageKey** – the message key used later to display message in appropriate language on UI.
- **params** – the params used later to change placeholders in the message

MdsException

public **MdsException** (*String* messageKey, *String...* params)

Constructs a new mds exception with the specified message key and params.

Parameters

- **messageKey** – the message key used later to display message in appropriate language on UI.
- **params** – the params used later to change placeholders in the message

MdsException

public **MdsException** (*String* messageKey, *Throwable* cause)

Constructs a new mds exception with the specified message key and the specified cause.

Parameters

- **messageKey** – the message key used later to display message in appropriate language on UI.
- **cause** – the cause of exception.

MdsException

public **MdsException** (*String* messageKey, *String* params, *Throwable* cause)

Constructs a new mds exception with the specified message key and the specified cause.

Parameters

- **messageKey** – the message key used later to display message in appropriate language on UI.
- **params** – the params used later to change placeholders in the message
- **cause** – the cause of exception.

Methods

getMessageKey

public *String* **getMessageKey** ()

getParams

```
public String getParams ()
```

7.42.18 MdsSchedulerException

```
public class MdsSchedulerException extends RuntimeException
```

The `MdsSchedulerException` exception signals problems with scheduling MDS jobs

Constructors

MdsSchedulerException

```
public MdsSchedulerException (String message, Throwable cause)
```

7.42.19 NoSuchTypeException

```
public class NoSuchTypeException extends MdsException
```

An exception which signals that a given type does not exist in the database.

Constructors

NoSuchTypeException

```
public NoSuchTypeException ()
```

Constructs a new `NoSuchTypeException` with `mds.error.noSuchType` as a message key.

7.42.20 ObjectNotFoundException

```
public class ObjectNotFoundException extends MdsException
```

Signals that the expected object was not found in the database.

Constructors

ObjectNotFoundException

```
public ObjectNotFoundException ()
```

7.42.21 ObjectReadException

```
public class ObjectReadException extends MdsException
```

Signals that we were unable to parse the object coming from the database.

Constructors

ObjectReadException

```
public ObjectReadException ()
```

ObjectReadException

```
public ObjectReadException (Throwable cause)
```

7.42.22 ObjectUpdateException

```
public class ObjectUpdateException extends MdsException
```

Signals that we were unable to update object instance from the provided data.

Constructors

ObjectUpdateException

```
public ObjectUpdateException ()
```

ObjectUpdateException

```
public ObjectUpdateException (Throwable cause)
```

7.42.23 ReservedKeywordException

```
public class ReservedKeywordException extends MdsException
```

Signals that field/lookup name is invalid because it is a java keyword.

Constructors

ReservedKeywordException

```
public ReservedKeywordException (String keyword)
```

7.42.24 SecurityException

```
public class SecurityException extends MdsException
```

The `SecurityException` exception signals a situation in which user wants to perform an operation on objects, they don't have access to.

Constructors

SecurityException

public **SecurityException** ()

Constructs a new SecurityException with *mds.error.securityError* as a message key.

7.42.25 ServiceNotFoundException

public class **ServiceNotFoundException** extends [MdsException](#)

Signals that service for a corresponding entity was not found. This most likely signals an issue with entities bundle.

Constructors

ServiceNotFoundException

public **ServiceNotFoundException** ()

7.42.26 TrashClassNotFoundException

public class **TrashClassNotFoundException** extends [MdsException](#)

An Exception thrown when MDS fails to load trash class for an entity

Constructors

TrashClassNotFoundException

public **TrashClassNotFoundException** ([String](#) *className*)

7.42.27 TypeAlreadyExistsException

public class **TypeAlreadyExistsException** extends [MdsException](#)

The `TypeAlreadyExistsException` is thrown, if the user attempts to add a field type, with a display name that already exists in the database

Constructors

TypeAlreadyExistsException

public **TypeAlreadyExistsException** ()

7.42.28 TypeNotFoundException

public class **TypeNotFoundException** extends [RuntimeException](#)

The `TypeNotFoundException` exception signals a situation in which a type with given name does not exist in database.

Constructors

TypeNotFoundException

```
public NotFoundException (String msg)
```

7.42.29 TypeValidationAlreadyExistsException

```
public class TypeValidationAlreadyExistsException extends MdsException
```

The `TypeValidationAlreadyExistsException` is thrown, if the user attempts to add a type validation when there is already validation for given type

Constructors

TypeValidationAlreadyExistsException

```
public TypeValidationAlreadyExistsException ()
```

7.43 org.motechproject.mds.filter

7.43.1 Filter

```
public class Filter implements Serializable
```

Represents a filter on a field.

Constructors

Filter

```
public Filter ()
```

Filter

```
public Filter (String field, FilterType type)
```

Methods

filterForQuery

```
public String filterForQuery ()
```

getField

```
public String getField ()
```

getType

```
public FilterType getType ()
```

paramsDeclarationForQuery

```
public String paramsDeclarationForQuery ()
```

requiresFiltering

```
public boolean requiresFiltering ()
```

setField

```
public void setField (String field)
```

setType

```
public void setType (FilterType type)
```

valuesForQuery

```
public Object[] valuesForQuery ()
```

7.43.2 [FilterType](#)

```
public enum FilterType  
    Represents a method of filtering.
```

Enum Constants**ALL**

```
public static final FilterType ALL
```

NO

```
public static final FilterType NO
```

PAST_7_DAYS

```
public static final FilterType PAST_7_DAYS
```

THIS_MONTH

public static final [FilterType](#) **THIS_MONTH**

THIS_YEAR

public static final [FilterType](#) **THIS_YEAR**

TODAY

public static final [FilterType](#) **TODAY**

YES

public static final [FilterType](#) **YES**

7.44 org.motechproject.mds.javassist

7.44.1 JavassistBuilder

public final class **JavassistBuilder**

Builder class for javassist related tasks. Helps with building appropriate elements of class e.g. fields, getters, field initializer

Methods

createEnumInitializer

public static CtField.Initializer **createEnumInitializer** ([String](#) *enumType*, [String](#) *defaultValue*)

createField

public static CtField **createField** (CtClass *declaring*, CtClass *type*, [String](#) *name*, [String](#) *genericSignature*)

createGetter

public static CtMethod **createGetter** ([String](#) *fieldName*, CtClass *declaring*, CtField *field*)

createInitializer

public static CtField.Initializer **createInitializer** ([String](#) *typeClass*, [String](#) *defaultValueAsString*)

createListInitializer

public static CtField.Initializer **createListInitializer** (*String genericType*, *Object defaultValue*)

createLocaleInitializer

public static CtField.Initializer **createLocaleInitializer** (*String defaultValue*)

createSetter

public static CtMethod **createSetter** (*String fieldName*, CtField *field*)

createSimpleInitializer

public static CtField.Initializer **createSimpleInitializer** (*String type*, *Object defaultValue*)

createSimpleInitializer

public static CtField.Initializer **createSimpleInitializer** (*String type*, *String defaultValue*)

getGetterName

public static *String* **getGetterName** (*String fieldName*, CtClass *declaring*, CtField *field*)

getSetterName

public static *String* **getSetterName** (*String fieldName*)

7.44.2 JavassistHelper

public final class **JavassistHelper**

Helper class for javassist related tasks. Helps with generic signature generation, plus methods related with analyzing and loading javassist class representations.

Methods

containsDeclaredField

public static boolean **containsDeclaredField** (CtClass *ctClass*, *String fieldName*)

containsDeclaredMethod

public static boolean **containsDeclaredMethod** (CtClass *ctClass*, *String methodName*)

containsField

public static boolean **containsField** (CtClass *ctClass*, String *fieldName*)

containsMethod

public static boolean **containsMethod** (CtClass *ctClass*, String *methodName*)

findDeclaredField

public static CtField **findDeclaredField** (CtClass *ctClass*, String *fieldName*)

findDeclaredMethod

public static CtMethod **findDeclaredMethod** (CtClass *ctClass*, String *methodName*)

findField

public static CtField **findField** (CtClass *ctClass*, String *fieldName*)

findMethod

public static CtMethod **findMethod** (CtClass *ctClass*, String *methodName*)

genericSignature

public static String **genericSignature** (Class<?> *typeClass*, Class<?> *genericParam*)

genericSignature

public static String **genericSignature** (Class<?> *typeClass*, String *genericParam*)

genericSignature

public static String **genericSignature** (String *typeClass*, String *genericParam*)

hasInterface

public static boolean **hasInterface** (CtClass *ctClass*, CtClass *ctInterface*)

inheritsFromCustomClass

public static boolean **inheritsFromCustomClass** (Class<?> *clazz*)

loadClass

public static CtClass **loadClass** (*Bundle bundle*, *String className*, *ClassPool classPool*)

removeDeclaredFieldIfExists

public static void **removeDeclaredFieldIfExists** (*CtClass ctClass*, *String fieldName*)

removeDeclaredMethodIfExists

public static void **removeDeclaredMethodIfExists** (*CtClass ctClass*, *String methodName*)

removeFieldIfExists

public static void **removeFieldIfExists** (*CtClass ctClass*, *String fieldName*)

removeMethodIfExists

public static void **removeMethodIfExists** (*CtClass ctClass*, *String methodName*)

toClassPath

public static *String* **toClassPath** (*Class<?> clazz*)

toClassPath

public static *String* **toClassPath** (*String clazz*)

toClassPath

public static *String* **toClassPath** (*String clazz*, boolean *extension*)

toGenericParam

public static *String* **toGenericParam** (*Class<?> clazz*)

toGenericParam

public static *String* **toGenericParam** (*String clazz*)

7.44.3 JavassistLoader

public class **JavassistLoader** extends [Loader<ClassData>](#)

The `JavassistLoader` is a implementation of the `org.motechproject.mds.util.Loader` interface. It takes class information from instance of `org.motechproject.mds.domain.ClassData` and the missing classes are taken from `org.motechproject.mds.javassist.MotechClassPool`

See also: `org.motechproject.mds.util.Loader`, `org.motechproject.mds.domain.ClassData`, `org.motechproject.mds.javassist.MotechClassPool`

Constructors

JavassistLoader

public **JavassistLoader** ([MDSClassLoader](#) *classLoader*)

Methods

doWhenClassNotFound

public void **doWhenClassNotFound** ([String](#) *name*)

getClassDefinition

public [Class](#)<?> **getClassDefinition** ([ClassData](#) *data*)

loadClass

public [Class](#)<?> **loadClass** ([ClassData](#) *arg*)

7.44.4 MotechClassPool

public final class **MotechClassPool**

This class holds the javasisst classpool, enriched by motech classes. All predefined additions to the `ClassPool` should take place here. The classpool should also be retrieved using this class, in order to be sure that the a initialization took place.

Methods

clearEnhancedData

public static void **clearEnhancedData** ()

getDefault

public static [ClassPool](#) **getDefault** ()

getEnhancedClassData

public static [ClassData](#) **getEnhancedClassData** ([String](#) *className*)

getEnhancedClasses

public static [Collection](#)<[ClassData](#)> **getEnhancedClasses** (boolean *includeInerfaces*)

getHistoryClassData

public static [ClassData](#) **getHistoryClassData** ([String](#) *className*)

getInterfaceName

public static [String](#) **getInterfaceName** ([String](#) *className*)

getRepositoryName

public static [String](#) **getRepositoryName** ([String](#) *className*)

getServiceImplName

public static [String](#) **getServiceImplName** ([String](#) *className*)

getServiceInterface

public static [String](#) **getServiceInterface** ([String](#) *className*)

getTrashClassData

public static [ClassData](#) **getTrashClassData** ([String](#) *className*)

isDDEReady

public static boolean **isDDEReady** ([String](#) *className*)

isServiceInterfaceRegistered

public static boolean **isServiceInterfaceRegistered** ([String](#) *className*)

registerDDE

public static void **registerDDE** ([String](#) *className*)

registerEnhancedClassData

```
public static void registerEnhancedClassData (ClassData enhancedClassData)
```

registerEnum

```
public static void registerEnum (String enumName)
```

registerHistoryClassData

```
public static void registerHistoryClassData (ClassData cData)
```

registerServiceInterface

```
public static void registerServiceInterface (String className, String interfaceName)
```

registerTrashClassData

```
public static void registerTrashClassData (ClassData cData)
```

registeredEnums

```
public static Collection<String> registeredEnums ()
```

registeredInterfaces

```
public static Collection<String> registeredInterfaces ()
```

unregisterEnhancedData

```
public static void unregisterEnhancedData (String className)
```

7.45 org.motechproject.mds.jdo

7.45.1 AbstractObjectValueGenerator

public abstract class **AbstractObjectValueGenerator**<T> implements ObjectValueGenerator

Base class for other generator classes. It takes value of property (see `getPropertyName()` method) from object and modify it depending on the implementation. If the modified value is null then the `java.lang.IllegalStateException` is thrown.

Parameters

- <T> – type of property

Methods

generate

public **Object** **generate** (ExecutionContext *ec*, **Object** *obj*, ExtensionMetaData[] *extensions*)

getPropertyName

protected abstract **String** **getPropertyName** ()

modify

protected abstract **T** **modify** (**T** *value*)

Modifies the given value. This method cannot return null value.

Parameters

- **value** – the given value related with property.

Returns modified value.

7.45.2 CreationDateValueGenerator

public class **CreationDateValueGenerator** extends **DateTimeValueGenerator**

The **CreationDateValueGenerator** class is responsible for generating value for `org.motechproject.mds.util.Constants.Util.CREATION_DATE_FIELD_NAME` field.

Methods

getPropertyName

protected **String** **getPropertyName** ()

7.45.3 CreatorValueGenerator

public class **CreatorValueGenerator** extends **UsernameValueGenerator**

The **CreatorValueGenerator** class is responsible for generating value for `org.motechproject.mds.util.Constants.Util.CREATOR_FIELD_NAME` field.

Methods

getPropertyName

protected **String** **getPropertyName** ()

7.45.4 DateTimeValueGenerator

public abstract class **DateTimeValueGenerator** extends [AbstractObjectValueGenerator<DateTime>](#)

The `DateTimeValueGenerator` class modifies properties with `org.joda.time.DateTime` type. If the given value is null then the current time is returned; otherwise the given value is returned.

Methods

modify

protected `DateTime` **modify** (`DateTime value`)

7.45.5 MDSCClassLoaderResolver

public class **MDSCClassLoaderResolver** implements `ClassLoaderResolver`

This is a wrapper for `org.motechproject.mds.jdo.MDSCClassLoaderResolverImpl`. All calls for the `org.datanucleus.ClassLoaderResolver` interface are passed to the current instance of the `ClassLoaderResolver` implementation. When we hit a `NullPointerException` originating in Felix, we can determine it is due to a synchronization bug after bundle updates - as a result of this `DataNucleus` has passed us `ClassLoaders` from the former Bundle version. In that case we reload the instance passing it the `ClassLoaders` from the new bundle.

Constructors

MDSCClassLoaderResolver

public **MDSCClassLoaderResolver** ()

MDSCClassLoaderResolver

public **MDSCClassLoaderResolver** (`ClassLoader pmLoader`)

Methods

classForName

public `Class` **classForName** (`String name`, `ClassLoader primary`)

classForName

public `Class` **classForName** (`String name`, `ClassLoader primary`, boolean *initialize*)

classForName

public `Class` **classForName** (`String name`)

classForName

public **Class** **classForName** (*String name*, boolean *initialize*)

getResource

public **URL** **getResource** (*String resourceName*, **ClassLoader** *primary*)

getResources

public **Enumeration**<**URL**> **getResources** (*String resourceName*, **ClassLoader** *primary*)

isAssignableFrom

public boolean **isAssignableFrom** (*String className*, **Class** *clazz*)

isAssignableFrom

public boolean **isAssignableFrom** (**Class** *clazz*, *String className*)

isAssignableFrom

public boolean **isAssignableFrom** (*String className1*, *String className2*)

registerUserClassLoader

public void **registerUserClassLoader** (**ClassLoader** *loader*)

setPrimary

public void **setPrimary** (**ClassLoader** *primary*)

setRuntimeClassLoader

public void **setRuntimeClassLoader** (**ClassLoader** *loader*)

unsetPrimary

public void **unsetPrimary** ()

7.45.6 MDSClassLoaderResolverImpl

class **MDSClassLoaderResolverImpl** extends `ClassLoaderResolverImpl`

The main purpose of the `MDSClassLoaderResolverImpl` class is to avoid situation in which standard datanucleus class loader resolver does not see classes that are saved in database. This is the main implementation that extends the standard `ClassLoaderResolverImpl` from datanucleus. Due to a synchronization bug in Felix, there are cases when we will instantiate this more then once (after we hit the bug).

Constructors

MDSClassLoaderResolverImpl

```
public MDSClassLoaderResolverImpl ()
```

MDSClassLoaderResolverImpl

```
public MDSClassLoaderResolverImpl (ClassLoader pmLoader)
```

Methods

classForName

```
public Class classForName (String name, ClassLoader primary)
```

7.45.7 MdsJdoAnnotationReader

public class **MdsJdoAnnotationReader** extends `JDOAnnotationReader`

MDS JDO annotation reader, extends the regular `org.datanucleus.api.jdo.metadata.JDOAnnotationReader`

This class was introduced because `org.datanucleus.api.jdo.metadata.JDOAnnotationReader` would not read field annotations for metadata if there was no class level JDO annotations. This extension will recognize the `org.motechproject.mds.annotations.Entity` annotation as an annotation indicating that the class is persistence capable.

Constructors

MdsJdoAnnotationReader

```
public MdsJdoAnnotationReader (MetaDataManager mgr)
```

Methods

isClassPersistenceCapable

```
protected AnnotationObject isClassPersistenceCapable (Class cls)
```

7.45.8 MdsTransactionManager

public class **MdsTransactionManager** extends [JdoTransactionManager](#)

We override springs transaction for classloader control. We store context classloaders as thread local variables, and switch them with the MDS classloader for the transaction. Since we only allow operations in transactions, this entry point for classloader switching is enough.

Methods

doBegin

protected void **doBegin** ([Object](#) *transaction*, [TransactionDefinition](#) *definition*)

doCleanupAfterCompletion

protected void **doCleanupAfterCompletion** ([Object](#) *transaction*)

7.45.9 ModificationDateValueGenerator

public class **ModificationDateValueGenerator** extends [DateTimeValueGenerator](#)

The `ModificationDateValueGenerator` class is responsible for generating value for `org.motechproject.mds.util.Constants.Util.MODIFICATION_DATE_FIELD_NAME` field.

Methods

getPropertyName

protected [String](#) **getPropertyName** ()

7.45.10 ModifiedByValueGenerator

public class **ModifiedByValueGenerator** extends [UsernameValueGenerator](#)

The `ModifiedByValueGenerator` class is responsible for generating value for `org.motechproject.mds.util.Constants.Util.MODIFIED_BY_FIELD_NAME` field.

Methods

getPropertyName

protected [String](#) **getPropertyName** ()

7.45.11 OwnerValueGenerator

public class **OwnerValueGenerator** extends [UsernameValueGenerator](#)

The `OwnerValueGenerator` class is responsible for generating value for `org.motechproject.mds.util.Constants.Util.OWNER_FIELD_NAME` field.

Methods

getPropertyName

protected `String` **getPropertyName** ()

7.45.12 SchemaGenerator

public class **SchemaGenerator** implements `InitializingBean`

The schema generator class is responsible for generating the table schema for entities upon start. Schema for all entity classes has to be generated, otherwise issues might arise in foreign key generation for example. This code runs in the generated entities bundle.

Constructors

SchemaGenerator

public **SchemaGenerator** (JDOPersistenceManagerFactory *persistenceManagerFactory*)

Methods

afterPropertiesSet

public void **afterPropertiesSet** ()

generateSchema

public void **generateSchema** ()

7.45.13 TimeTypeConverter

public class **TimeTypeConverter** implements `TypeConverter<Time, String>`

This is datanucleus type converter we plug in. It is responsible for converting `org.motechproject.commons.date.model.Time` instances to Strings which are persisted.

Methods

toDatastoreType

public `String` **toDatastoreType** (`Time` *memberValue*)

toMemberType

public `Time` **toMemberType** (`String` *datastoreValue*)

7.45.14 UsernameValueGenerator

public abstract class **UsernameValueGenerator** extends [AbstractObjectValueGenerator<String>](#)

The `UsernameValueGenerator` class modifies properties with `java.lang.String` type. The given value is returned without any change if it is not blank. Otherwise the class tries to get current logged user name. If the user exists and name is not blank then this name is returned otherwise the empty string is returned.

Methods

modify

protected [String](#) **modify** ([String](#) *value*)

7.46 org.motechproject.mds.query

7.46.1 CollectionProperty

public class **CollectionProperty** extends [Property<Collection>](#)

The `CollectionProperty` class represent a property that will be used in JDO query and it has to have the given value(s).

Constructors

CollectionProperty

public **CollectionProperty** ([String](#) *name*, [Object](#) *value*)

Methods

generateDeclareParameter

public [CharSequence](#) **generateDeclareParameter** (int *idx*)

generateFilter

public [CharSequence](#) **generateFilter** (int *idx*)

shouldIgnoreThisProperty

protected boolean **shouldIgnoreThisProperty** ()

unwrap

public [Collection](#) **unwrap** ()

7.46.2 CustomOperatorProperty

public class **CustomOperatorProperty**<T> extends [Property](#)<T>

The `CustomOperatorProperty` class represents a property that will be used in JDO query. This class allows inserting a custom operator, such as `>`, `<=`, `matches()`, etc.

Parameters

- <T> – type of the passed value

Constructors

CustomOperatorProperty

public **CustomOperatorProperty** ([String](#) name, T value, [String](#) operator)

Methods

generateFilter

public [CharSequence](#) **generateFilter** (int idx)

getOperator

public [String](#) **getOperator** ()

isOperatorAMethod

public boolean **isOperatorAMethod** ()

7.46.3 EqualProperty

public class **EqualProperty**<T> extends [Property](#)<T>

The `EqualProperty` class represents a property that will be used in JDO query and it has to be equal to the given value.

Parameters

- <T> – type of the passed value

Constructors

EqualProperty

public **EqualProperty** ([String](#) name, T value)

Methods

generateFilter

```
public CharSequence generateFilter (int idx)
```

7.46.4 MatchesProperty

```
public class MatchesProperty extends CustomOperatorProperty<String>
```

A convenience extension of the `org.motechproject.mds.query.CustomOperatorProperty`.

The custom operator is “matches()” and the underlying type is String. The value passed will be wrapped inside `.*.*` for matching purposes.

Constructors

MatchesProperty

```
public MatchesProperty (String name, String value)
```

7.46.5 Property

```
public abstract class Property<T>
```

The `Property` class represents a property that will be used in JDO query. Classes that extend this class should define how that property should be used in WHERE section in JDO query.

Parameters

- `<T>` – type of the passed value

Constructors

Property

```
protected Property (String name, T value)
```

Methods

asDeclareParameter

```
public CharSequence asDeclareParameter (int idx)
```

asFilter

```
public CharSequence asFilter (int idx)
```

containsOnlyNullValues

```
protected boolean containsOnlyNullValues (Collection collection)
```

generateDeclareParameter

```
protected CharSequence generateDeclareParameter (int idx)
```

generateFilter

```
protected abstract CharSequence generateFilter (int idx)
```

getName

```
public String getName ()
```

getValue

```
public T getValue ()
```

shouldIgnoreThisProperty

```
protected boolean shouldIgnoreThisProperty ()
```

unwrap

```
public Collection unwrap ()
```

7.46.6 PropertyBuilder

```
public final class PropertyBuilder
```

The `PropertyBuilder` class is a util class that helps create appropriate property class based on passed name and value.

Methods**create**

```
public static Property create (Field field, Object value)
```

create

```
public static Property create (String name, Object value)
```

create

```
public static Property create (String name, Object value, String operator)
```

7.46.7 QueryExecution

public interface **QueryExecution**<T>

Allows users to execute custom queries through Motech Data Services. Implementations need only to implement the execute method, which can operate directly on the `javax.jdo.Query` object. The return value type is left to the implementation.

Parameters

- <T> – the type that will be returned from this query

Methods

execute

T **execute** (Query *query*, InstanceSecurityRestriction *restriction*)

7.46.8 QueryExecutor

public final class **QueryExecutor**

The `QueryExecutor` util class provides methods that help execute a JDO query.

Methods

execute

public static Object **execute** (Query *query*, InstanceSecurityRestriction *restriction*)

execute

public static Object **execute** (Query *query*, Object *value*, InstanceSecurityRestriction *restriction*)

executeWithArray

public static Object **executeWithArray** (Query *query*, Object[] *values*, InstanceSecurityRestriction *restriction*)

executeWithArray

public static Object **executeWithArray** (Query *query*, List<Property> *properties*)

executeWithFilter

public static Object **executeWithFilter** (Query *query*, Filter *filter*, InstanceSecurityRestriction *restriction*)

7.46.9 QueryParams

public class **QueryParams** implements [Serializable](#)

Utility class containing parameters which control order and size of query results. Used mainly for paging/ordering queries from the UI.

Constructors

QueryParams

public **QueryParams** ([Integer](#) *page*, [Integer](#) *pageSize*)

QueryParams

public **QueryParams** ([Order](#) *order*)

QueryParams

public **QueryParams** ([Integer](#) *page*, [Integer](#) *pageSize*, [Order](#) *order*)

Methods

ascOrder

public static [QueryParams](#) **ascOrder** ([String](#) *field*)

descOrder

public static [QueryParams](#) **descOrder** ([String](#) *field*)

getOrder

public [Order](#) **getOrder** ()

getPage

public [Integer](#) **getPage** ()

getPageSize

public [Integer](#) **getPageSize** ()

isOrderSet

public boolean **isOrderSet** ()

isPagingSet

public boolean **isPagingSet** ()

7.46.10 QueryUtil

public final class **QueryUtil**

The `QueryUtil` util class provides methods that help developer to create a JDO query.

See also: `javax.jdo.Query`

Methods

asMatchesPattern

public static `String` **asMatchesPattern** (`String` *string*)

Returns the string in a form a pattern used for searching with the matches operator.

Parameters

- **string** – The string to search for.

Returns The string in format `.*<string>.*` or the string unchanged if it is empty.

setCountResult

public static void **setCountResult** (`Query` *query*)

setQueryParams

public static void **setQueryParams** (`Query` *query*, `QueryParams` *queryParams*)

useFilter

public static void **useFilter** (`Query` *query*, `Filter` *filter*)

useFilter

public static void **useFilter** (`Query` *query*, `String`[] *properties*, `Object`[] *values*)

useFilter

public static void **useFilter** (`Query` *query*, `String`[] *properties*, `Object`[] *values*, `InstanceSecurityRestriction` *restriction*)

useFilter

public static void **useFilter** (`Query` *query*, `List`<`Property`> *properties*)

useFilter

```
public static void useFilter (Query query, List<Property> properties, InstanceSecurityRestriction restriction)
```

useFilterFromPattern

```
public static void useFilterFromPattern (Query query, String pattern, List<Property> properties)
```

7.46.11 RangeProperty

```
public class RangeProperty<T> extends Property<Range<T>>
```

The `RangeProperty` class represents a property that will be used in JDO query and it has to be inside the given range.

Parameters

- `<T>` – type used in range.

Constructors

RangeProperty

```
public RangeProperty (String name, Range<T> value)
```

Methods

generateDeclareParameter

```
public CharSequence generateDeclareParameter (int idx)
```

generateFilter

```
public CharSequence generateFilter (int idx)
```

shouldIgnoreThisProperty

```
protected boolean shouldIgnoreThisProperty ()
```

unwrap

```
public Collection unwrap ()
```

7.46.12 RestrictionProperty

```
public class RestrictionProperty extends EqualProperty<String>
```

The `RestrictionProperty` class represents a property that will be used in JDO query and depends on restriction criteria the `creator` or `owner` field in an instance has to have the appropriate user name.

Constructors

RestrictionProperty

```
public RestrictionProperty (InstanceSecurityRestriction restriction, String value)
```

7.46.13 SetProperty

```
public class SetProperty<T> extends Property<Set<T>>
```

The `SetProperty` class represents a property that will be used in JDO query and it has to have one of the value from the given set.

Parameters

- `<T>` – type used in set.

Constructors

SetProperty

```
public SetProperty (String name, Set<T> value)
```

Methods

generateDeclareParameter

```
public CharSequence generateDeclareParameter (int idx)
```

generateFilter

```
public CharSequence generateFilter (int idx)
```

shouldIgnoreThisProperty

```
protected boolean shouldIgnoreThisProperty ()
```

unwrap

```
public Collection unwrap ()
```

7.47 org.motechproject.mds.repository

7.47.1 AllConfigSettings

```
public class AllConfigSettings extends MotechDataRepository<ConfigSettings>
```

`AllConfigSettings` is responsible for communication with database for MDS configuration.

Constructors

AllConfigSettings

```
public AllConfigSettings ()
```

Methods

addOrUpdate

```
public void addOrUpdate (ConfigSettings record)
```

7.47.2 AllEntities

```
public class AllEntities extends MotechDataRepository<Entity>
```

The `AllEntities` class is a repository class that operates on instances of `org.motechproject.mds.domain.Entity`.

Constructors

AllEntities

```
public AllEntities ()
```

Methods

contains

```
public boolean contains (String className)
```

create

```
public Entity create (EntityDto dto)
```

delete

```
public void delete (Long id)
```

retrieveByClassName

```
public Entity retrieveByClassName (String className)
```

retrieveById

```
public Entity retrieveById (Long id)
```

7.47.3 AllEntityAudits

public class **AllEntityAudits** extends [MotechDataRepository<EntityAudit>](#)

This a repository for persisting entity audits. It provides methods which created audits, by cloning the giving entity.

Constructors

AllEntityAudits

public **AllEntityAudits** ()

Methods

createAudit

public [EntityAudit](#) **createAudit** ([Entity](#) entity, [String](#) username)

7.47.4 AllEntityDrafts

public class **AllEntityDrafts** extends [MotechDataRepository<EntityDraft>](#)

This a repository for persisting entity drafts. It provides methods which created drafts, by cloning the giving entity.

Constructors

AllEntityDrafts

public **AllEntityDrafts** ()

Methods

create

public [EntityDraft](#) **create** ([Entity](#) entity, [String](#) username)

deleteAll

public void **deleteAll** ([Entity](#) entity)

retrieve

public [EntityDraft](#) **retrieve** ([Entity](#) entity, [String](#) username)

retrieveAll

```
public List<EntityDraft> retrieveAll (String username)
```

retrieveAll

```
public List<EntityDraft> retrieveAll (Entity entity)
```

setProperties

```
public void setProperties (EntityDraft draft, Entity entity)
```

setProperties

```
public void setProperties (EntityDraft draft, Entity entity, String username)
```

update

```
public EntityDraft update (EntityDraft draft)
```

7.47.5 AllTypeSettings

```
public class AllTypeSettings extends MotechDataRepository<TypeSetting>
```

The AllTypeSettings class is a repository class that operates on instances of `org.motechproject.mds.domain.TypeSetting`.

Constructors**AllTypeSettings**

```
public AllTypeSettings ()
```

7.47.6 AllTypeValidations

```
public class AllTypeValidations extends MotechDataRepository<TypeValidation>
```

The AllTypeValidations class is a repository class that operates on instances of `org.motechproject.mds.domain.TypeValidation`.

Constructors**AllTypeValidations**

```
public AllTypeValidations ()
```

7.47.7 AllTypes

public class **AllTypes** extends [MotechDataRepository<Type>](#)

The `AllTypes` repository class allows persistence and retrieving of Field Types in Data Services database.

Constructors

AllTypes

public **AllTypes** ()

Methods

retrieveByClassName

public [Type](#) **retrieveByClassName** ([String](#) *className*)

7.47.8 MetadataHolder

public class **MetadataHolder**

Holds the current JDO metadata for Seuss. Allows reloading the metadata and retrieval for modifications.

Methods

addProcessedRelation

public void **addProcessedRelation** ([String](#) *className*)

getJdoMetadata

public [JDOMetadata](#) **getJdoMetadata** ()

isRelationProcessed

public boolean **isRelationProcessed** ([String](#) *className*)

reloadMetadata

public [JDOMetadata](#) **reloadMetadata** ()

setPersistenceManagerFactory

public void **setPersistenceManagerFactory** ([PersistenceManagerFactory](#) *persistenceManagerFactory*)

7.47.9 MotechDataRepository

public abstract class **MotechDataRepository**<T>

This is a basic repository class with standard CRUD operations. It should be used by other repositories inside this package.

This class is also used as super class to create a repository related with the given entity schema in `org.motechproject.mds.builder.EntityInfrastructureBuilder`.

Parameters

- <T> – the type of class

Constructors

MotechDataRepository

protected **MotechDataRepository** ([Class](#)<T> *classType*)

Methods

count

public long **count** ([InstanceSecurityRestriction](#) *restriction*)

count

public long **count** ([String](#)[] *properties*, [Object](#)[] *values*, [InstanceSecurityRestriction](#) *restriction*)

count

public long **count** ([List](#)<[Property](#)> *properties*, [InstanceSecurityRestriction](#) *restriction*)

countForFilter

public long **countForFilter** ([Filter](#) *filter*)

countForFilter

public long **countForFilter** ([Filter](#) *filter*, [InstanceSecurityRestriction](#) *restriction*)

create

public T **create** (T *object*)

delete

public void **delete** (T *object*)

delete

public void **delete** (*String* *property*, *Object* *value*, *InstanceSecurityRestriction* *restriction*)

delete

public void **delete** (*String*[] *properties*, *Object*[] *values*, *InstanceSecurityRestriction* *restriction*)

deleteAll

public void **deleteAll** (*String* *property*, *Object* *value*)

deleteAll

public void **deleteAll** (*String* *property*, *Object* *value*, *InstanceSecurityRestriction* *restriction*)

deleteAll

public void **deleteAll** (*String*[] *properties*, *Object*[] *values*, *InstanceSecurityRestriction* *restriction*)

exists

public boolean **exists** (*String* *property*, *Object* *value*)

exists

public boolean **exists** (*String*[] *properties*, *Object*[] *values*)

filter

public *List*<*T*> **filter** (*Filter* *filter*, *QueryParams* *queryParams*)

filter

public *List*<*T*> **filter** (*Filter* *filter*, *QueryParams* *queryParams*, *InstanceSecurityRestriction* *restriction*)

getClassType

public *Class*<*T*> **getClassType** ()

getDetachedField

public *Object* **getDetachedField** (*T* *instance*, *String* *field*)

getPersistenceManager

```
public PersistenceManager getPersistenceManager ()
```

retrieve

```
public T retrieve (Object key)
```

retrieve

```
public T retrieve (String property, Object value)
```

retrieve

```
public T retrieve (String property, Object value, InstanceSecurityRestriction restriction)
```

retrieve

```
public T retrieve (String[] properties, Object[] values)
```

retrieve

```
public T retrieve (String[] properties, Object[] values, InstanceSecurityRestriction restriction)
```

retrieveAll

```
public List<T> retrieveAll ()
```

retrieveAll

```
public List<T> retrieveAll (InstanceSecurityRestriction restriction)
```

retrieveAll

```
public List<T> retrieveAll (String property, Object value)
```

retrieveAll

```
public List<T> retrieveAll (String property, Object value, InstanceSecurityRestriction restriction)
```

retrieveAll

```
public List<T> retrieveAll (String[] properties, Object[] values, InstanceSecurityRestriction restriction)
```

retrieveAll

```
public List<T> retrieveAll (String[] properties, Object[] values, QueryParams queryParams, InstanceSecurityRestriction restriction)
```

retrieveAll

```
public List<T> retrieveAll (QueryParams queryParams, InstanceSecurityRestriction restriction)
```

retrieveAll

```
public List<T> retrieveAll (List<Property> properties, InstanceSecurityRestriction restriction)
```

retrieveAll

```
public List<T> retrieveAll (List<Property> properties, QueryParams queryParams, InstanceSecurityRestriction restriction)
```

setPersistenceManagerFactory

```
public void setPersistenceManagerFactory (PersistenceManagerFactory persistenceManagerFactory)
```

update

```
public T update (T object)
```

7.48 org.motechproject.mds.service

7.48.1 DefaultMotechDataService

public abstract class **DefaultMotechDataService**<T> implements **MotechDataService**<T>

This is a basic implementation of `org.motechproject.mds.service.MotechDataService`. Mainly it is used as super class to create a service related with the given entity schema in `org.motechproject.mds.builder.EntityInfrastructureBuilder` but it can be also used by other services inside this package.

Parameters

- <T> – the type of entity schema.

Methods**count**

```
public long count ()
```

count

protected long **count** ([List](#)<[Property](#)> *properties*)

countForFilter

public long **countForFilter** ([Filter](#) *filter*)

create

public T **create** (T *object*)

delete

public void **delete** (T *object*)

delete

public void **delete** ([String](#) *primaryKeyName*, [Object](#) *value*)

deleteAll

public void **deleteAll** ()

doInTransaction

public <R> R **doInTransaction** ([TransactionCallback](#)<R> *transactionCallback*)

executeQuery

public <R> R **executeQuery** ([QueryExecution](#)<R> *queryExecution*)

filter

public [List](#)<T> **filter** ([Filter](#) *filter*)

filter

public [List](#)<T> **filter** ([Filter](#) *filter*, [QueryParams](#) *queryParams*)

findById

public T **findById** ([Long](#) *id*)

findTrashInstanceById

public T **findTrashInstanceById** (Object *instanceId*, Object *entityId*)

getDetachedField

public Object **getDetachedField** (T *instance*, String *fieldName*)

getId

protected Object **getId** (T *instance*)

getRepository

protected MotechDataRepository<T> **getRepository** ()

initializeSecurityState

public void **initializeSecurityState** ()

retrieve

public T **retrieve** (String *primaryKeyName*, Object *value*)

retrieveAll

public List<T> **retrieveAll** ()

retrieveAll

public List<T> **retrieveAll** (QueryParams *queryParams*)

retrieveAll

protected List<T> **retrieveAll** (List<Property> *properties*)

retrieveAll

protected List<T> **retrieveAll** (List<Property> *properties*, QueryParams *queryParams*)

revertFromTrash

public void **revertFromTrash** (Object *newInstance*, Object *trash*)

setAllEntities

public void **setAllEntities** (*AllEntities allEntities*)

setEntityService

public void **setEntityService** (*EntityService entityService*)

setHistoryService

public void **setHistoryService** (*HistoryService historyService*)

setRepository

public void **setRepository** (*MotechDataRepository<T> repository*)

setTransactionManager

public void **setTransactionManager** (*JdoTransactionManager transactionManager*)

setTrashService

public void **setTrashService** (*TrashService trashService*)

update

public T **update** (T *object*)

validateCredentials

protected *InstanceSecurityRestriction* **validateCredentials** ()

validateCredentials

protected *InstanceSecurityRestriction* **validateCredentials** (T *instance*)

7.48.2 EntityService

public interface **EntityService**

This interface provides methods related with executing actions on an entity.

Methods

abandonChanges

void **abandonChanges** (*Long entityId*)

addDisplayedFields

void **addDisplayedFields** (*EntityDto entityDto*, *Map<String, Long> positions*)

addFields

void **addFields** (*EntityDto entity*, *Collection<FieldDto> fields*)

addFilterableFields

void **addFilterableFields** (*EntityDto entityDto*, *Collection<String> fieldNames*)

addLookups

void **addLookups** (*Long entityId*, *Collection<LookupDto> lookups*)

commitChanges

void **commitChanges** (*Long entityId*)

commitChanges

void **commitChanges** (*Long entityId*, *String changesOwner*)

createEntity

EntityDto **createEntity** (*EntityDto entityDto*)

deleteEntity

void **deleteEntity** (*Long entityId*)

findEntityFieldByName

FieldDto **findEntityFieldByName** (*Long entityId*, *String name*)

findFieldByName

FieldDto **findFieldByName** (*Long entityId*, *String name*)

getAdvancedSettings

AdvancedSettingsDto **getAdvancedSettings** (Long *entityId*)

getAdvancedSettings

AdvancedSettingsDto **getAdvancedSettings** (Long *entityId*, boolean *committed*)

getCurrentSchemaVersion

Long **getCurrentSchemaVersion** (String *entityClassName*)

getDisplayFields

List<FieldDto> **getDisplayFields** (Long *entityId*)

getEntitiesWithLookups

List<EntityDto> **getEntitiesWithLookups** ()

getEntity

EntityDto **getEntity** (Long *entityId*)

getEntityByClassName

EntityDto **getEntityByClassName** (String *className*)

getEntityDraft

EntityDraft **getEntityDraft** (Long *entityId*)

getEntityDraft

EntityDraft **getEntityDraft** (Long *entityId*, String *username*)

getEntityFields

List<FieldDto> **getEntityFields** (Long *entityId*)

getEntityForEdit

EntityDto **getEntityForEdit** (Long *entityId*)

getEntityLookups

List<LookupDto> **getEntityLookups** (Long *entityId*)

getFields

List<FieldDto> **getFields** (Long *entityId*)

getLookupByName

LookupDto **getLookupByName** (Long *entityId*, String *lookupName*)

listEntities

List<EntityDto> **listEntities** ()

listEntities

List<EntityDto> **listEntities** (boolean *withSecurityCheck*)

listWorkInProgress

List<EntityDto> **listWorkInProgress** ()

saveDraftEntityChanges

DraftResult **saveDraftEntityChanges** (Long *entityId*, DraftData *draftData*, String *username*)

saveDraftEntityChanges

DraftResult **saveDraftEntityChanges** (Long *entityId*, DraftData *draftData*)

updateComboboxValues

void **updateComboboxValues** (Long *entityId*, Map<String, Collection> *fieldValuesToUpdate*)

updateDraft

EntityDto **updateDraft** (Long *entityId*)

7.48.3 HistoryService

public interface **HistoryService**

The **HistoryService** provides methods related with processing historical changes on the given instance of entity.

Methods

countHistoryRecords

long **countHistoryRecords** (*Object instance*)

getHistoryForInstance

List **getHistoryForInstance** (*Object instance*, *QueryParams queryParams*)

Returns the historical data for the given instance. This method return historical data only for objects that are not in the MDS trash. For trash instances the return value will be incorrect.

Parameters

- **instance** – an instance created from the given entity definition.
- **queryParams** – Query parameters such as page number, size of page and sort direction. If null method will return all history records.

Returns a list of historical data related with the given instance.

record

void **record** (*Object instance*)

Records changes made on the given instance of entity. The first historical data should be equal to data inside the given instance. Two instance of historical data should be connected using appropriate fields (defined in history class definition). This method should be used only for instances that are not in the MDS trash.

Parameters

- **instance** – an instance created from the given entity definition.

remove

void **remove** (*Object instance*)

Removes all historical data with information what changes were made on the given instance of entity.

Parameters

- **instance** – an instance created from the given entity definition.

setTrashFlag

void **setTrashFlag** (*Object instance*, *Object trash*, boolean *flag*)

Sets the trash flag for historical data related with the given instance object.

Parameters

- **instance** – an instance created from the given entity definition.
- **flag** – true if instance was moved to trash; otherwise false.

7.48.4 JarGeneratorService

public interface **JarGeneratorService**

This interface provides methods to create a bundle jar with all entities defined in MDS module.

Fields

BLUEPRINT_TEMPLATE

String **BLUEPRINT_TEMPLATE**

BLUEPRINT_XML

String **BLUEPRINT_XML**

BUNDLE_IMPORTS

String **BUNDLE_IMPORTS**

DATANUCLEUS_PROPERTIES

String **DATANUCLEUS_PROPERTIES**

ENTITY_LIST_FILE

String **ENTITY_LIST_FILE**

MDS_COMMON_CONTEXT

String **MDS_COMMON_CONTEXT**

MDS_ENTITIES_CONTEXT

String **MDS_ENTITIES_CONTEXT**

MDS_ENTITIES_CONTEXT_TEMPLATE

String **MDS_ENTITIES_CONTEXT_TEMPLATE**

MOTECH_MDS_PROPERTIES

String **MOTECH_MDS_PROPERTIES**

PACKAGE_JDO

String **PACKAGE_JDO**

Methods

generate

File **generate** ()

Generates a jar file that contains entity class definitions, repositories, interfaces, implementations of these interfaces. The jar should also contains class related with historical data and trash.

Throws

- **IOException** – if an I/O error occurs while the jar is creating.

Returns file that point to an entitites bundle jar.

regenerateMdsDataBundle

void **regenerateMdsDataBundle** (boolean *buildDDE*)

Constructs entities, builds and starts the entities bundle jar

Parameters

- **buildDDE** – `true` if class definitions for entities from outside bundles should also be created; otherwise `false`.

See also: `.generate()`

regenerateMdsDataBundle

void **regenerateMdsDataBundle** (boolean *buildDDE*, boolean *startBundle*)

Constructs entities, builds the entities bundle jar. The generated bundle will start only if the **startBundle** will be set to `true`.

Parameters

- **buildDDE** – `true` if class definitions for entities from outside bundles should also be created; otherwise `false`.
- **startBundle** – `true` if the generated bundle should start; otherwise `false`.

See also: `.generate()`

regenerateMdsDataBundleAfterDdeEnhancement

void **regenerateMdsDataBundleAfterDdeEnhancement** (String *moduleName*)

Constructs entities, builds and starts the entities bundle jar. This method should be used after DDE enhancement. It will build all DDE classes and refresh the module from which the DDE being enhanced comes from.

Parameters

- **moduleName** – module name of the entity from which the enhanced DDE comes from

See also: `.generate()`

7.48.5 MdsSchedulerService

public interface **MdsSchedulerService**

Methods

scheduleRepeatingJob

void **scheduleRepeatingJob** (long *interval*)

unscheduleRepeatingJob

void **unscheduleRepeatingJob** ()

7.48.6 MotechDataService

public interface **MotechDataService**<T>

This is a basic service interface with CRUD operations. Mainly it is used as super interface to create service interface related with the given entity schema in [org.motechproject.mds.builder.EntityInfrastructureBuilder](#) but it can be also used by other service interfaces inside this package.

Parameters

- <T> – the type of entity schema.

Methods

count

long **count** ()

countForFilter

long **countForFilter** ([Filter](#) *filter*)

create

T **create** (T *object*)

delete

void **delete** (T *object*)

delete

void **delete** ([String](#) *primaryKeyName*, [Object](#) *value*)

deleteAll

void **deleteAll** ()

doInTransaction

<R> R **doInTransaction** (TransactionCallback<R> *transactionCallback*)

executeQuery

<R> R **executeQuery** (QueryExecution<R> *queryExecution*)

filter

List<T> **filter** (Filter *filter*)

filter

List<T> **filter** (Filter *filter*, QueryParams *queryParams*)

findById

T **findById** (Long *id*)

findTrashInstanceById

T **findTrashInstanceById** (Object *instanceId*, Object *entityId*)

getDetachedField

Object **getDetachedField** (T *instance*, String *fieldName*)

retrieve

T **retrieve** (String *primaryKeyName*, Object *value*)

retrieveAll

List<T> **retrieveAll** ()

retrieveAll

List<T> **retrieveAll** (QueryParams *queryParams*)

revertFromTrash

void **revertFromTrash** (*Object newInstance*, *Object trash*)

update

T **update** (T *object*)

7.48.7 TransactionalMotechDataService

public abstract class **TransactionalMotechDataService**<T> extends **DefaultMotechDataService**<T>

The main goal of the **TransactionalMotechDataService** class is to resolve problems with transaction annotations not working for generated lookups. We use the traditional transaction callback instead.

Parameters

- <T> – the type of entity schema.

Methods**count**

protected long **count** (*List*<*Property*> *properties*)

retrieveAll

protected *List*<T> **retrieveAll** (*List*<*Property*> *properties*)

retrieveAll

protected *List*<T> **retrieveAll** (*List*<*Property*> *properties*, *QueryParams queryParams*)

7.48.8 TrashService

public interface **TrashService**

The **TrashService** provides methods related with the module trash mode (by default the mode is active and it can be turned off by the user).

Methods**countTrashRecords**

long **countTrashRecords** (*String className*)

emptyTrash

void **emptyTrash** ()

Cleans the module trash. All instances in trash should be removed permanently and if they contain any historical data they should also be removed permanently.

This method should only be executed by the job created in the `scheduleEmptyTrashJob()` method.

findTrashById

Object **findTrashById** (Object *instanceId*, Object *entityId*)

Return instance with given id from trash.

Parameters

- **instanceId** – id of instance
- **entityId** – id of instance entity

getInstancesFromTrash

Collection **getInstancesFromTrash** (String *entityName*, QueryParams *queryParams*)

Returns the collection of instances from trash of a certain entity. Returned collection contains only instances that are on the current schema version.

Parameters

- **entityName** – Instances of what entity should be looked for
- **queryParams** – Query parameters such as page number, size of page and sort direction. If null method will return all records in trash.

Returns Collection of instances on the current schema version in trash

isTrashMode

boolean **isTrashMode** ()

Checks if trash mode is active. This method should be used before executing the `moveToTrash(Object, Long)` method to resolve whether the given instance should be moved to trash or removed permanently.

Returns true if delete mode is equal to `org.motechproject.mds.config.DeleteMode.TRASH`; false otherwise.

moveFromTrash

void **moveFromTrash** (Object *newInstance*, Object *trash*)

Sets history for given trashed instance to match the new one and deletes trashed one from trash.

Parameters

- **newInstance** – instance to be returned from trash
- **trash** – trashed instance to be removed

moveToTrash

void **moveToTrash** (*Object instance*, *Long schemaVersion*)

Moves the given instance to the trash. This method should only be executed, when the module trash mode is active.

Parameters

- **instance** – an instance created from the given entity definition.

See also: `.isTrashMode()`

scheduleEmptyTrashJob

void **scheduleEmptyTrashJob** ()

Sets the repeating schedule job that will be executed from time to time. Execution time depends on the value of time value and time unit (defined in `org.motechproject.mds.util.Constants.Config.MODULE_FILE`).

Before scheduling new job, the old one should be unscheduled to prevent the errors.

7.48.9 TypeService

public interface **TypeService**

The `TypeService` is an interface defining available methods to execute various actions on Field Types.

Methods

findType

TypeDto **findType** (*Class<?> clazz*)

findValidations

List<TypeValidation> **findValidations** (*TypeDto type*, *Class<? extends Annotation> aClass*)

getAllTypes

List<TypeDto> **getAllTypes** ()

getType

Type **getType** (*TypeValidation validation*)

7.49 org.motechproject.mds.service.impl

7.49.1 BasePersistenceService

public abstract class **BasePersistenceService**

The `BasePersistenceService` class provides utility methods for communication with the database for `HistoryServiceImpl` and `TrashServiceImpl`. It allows to create and retrieve instances, load proper classes and parse values.

Methods

create

protected <T> **Object create** (*Class*<T> *clazz*, *Object src*, *EntityType type*)

create

protected <T> **Object create** (*Class*<T> *clazz*, *Object src*, *EntityType type*, *ObjectReference objectReference*)

findService

protected *MotechDataService* **findService** (*Class*<?> *clazz*)

getClass

protected *Class*<?> **getClass** (*Object src*, *EntityType type*)

getClass

protected *Class*<?> **getClass** (*String srcClassName*, *EntityType type*)

getCurrentSchemaVersion

protected *Long* **getCurrentSchemaVersion** (*String className*)

getEntities

protected *List*<*Entity*> **getEntities** ()

getEntity

protected *Entity* **getEntity** (*Long id*)

getEntitySchemaVersion

protected [Long](#) **getEntitySchemaVersion** ([Object](#) *src*)

getInstanceClassName

protected [String](#) **getInstanceClassName** ([Object](#) *instance*)

getInstanceId

protected [Long](#) **getInstanceId** ([Object](#) *instance*)

getPersistenceManagerFactory

protected [PersistenceManagerFactory](#) **getPersistenceManagerFactory** ()

getValue

protected [Object](#) **getValue** ([Field](#) *field*, [Object](#) *src*, [Object](#) *target*, [EntityType](#) *type*, [ObjectReference](#) *objectReference*)

setAllEntities

public void **setAllEntities** ([AllEntities](#) *allEntities*)

setBundleContext

public void **setBundleContext** ([BundleContext](#) *bundleContext*)

setPersistenceManagerFactory

public void **setPersistenceManagerFactory** ([PersistenceManagerFactory](#) *persistenceManagerFactory*)

7.49.2 EntityServiceImpl

public class **EntityServiceImpl** implements [EntityService](#)

Default implementation of [org.motechproject.mds.service.EntityService](#) interface.

Methods

abandonChanges

public void **abandonChanges** ([Long](#) *entityId*)

addDisplayedFields

```
public void addDisplayedFields (EntityDto entityDto, Map<String, Long> positions)
```

addFields

```
public void addFields (EntityDto entityDto, Collection<FieldDto> fields)
```

addFilterableFields

```
public void addFilterableFields (EntityDto entityDto, Collection<String> fieldNames)
```

addLookups

```
public void addLookups (Long entityId, Collection<LookupDto> lookups)
```

commitChanges

```
public void commitChanges (Long entityId, String changesOwner)
```

commitChanges

```
public void commitChanges (Long entityId)
```

createEntity

```
public EntityDto createEntity (EntityDto entityDto)
```

deleteEntity

```
public void deleteEntity (Long entityId)
```

findEntityFieldByName

```
public FieldDto findEntityFieldByName (Long entityId, String name)
```

findFieldByName

```
public FieldDto findFieldByName (Long entityId, String name)
```

getAdvancedSettings

```
public AdvancedSettingsDto getAdvancedSettings (Long entityId)
```

getAdvancedSettings

public [AdvancedSettingsDto](#) **getAdvancedSettings** ([Long](#) *entityId*, boolean *committed*)

getCurrentSchemaVersion

public [Long](#) **getCurrentSchemaVersion** ([String](#) *className*)

getDisplayFields

public [List](#)<[FieldDto](#)> **getDisplayFields** ([Long](#) *entityId*)

getEntitiesWithLookups

public [List](#)<[EntityDto](#)> **getEntitiesWithLookups** ()

getEntity

public [EntityDto](#) **getEntity** ([Long](#) *entityId*)

getEntityByClassName

public [EntityDto](#) **getEntityByClassName** ([String](#) *className*)

getEntityDraft

public [EntityDraft](#) **getEntityDraft** ([Long](#) *entityId*)

getEntityDraft

public [EntityDraft](#) **getEntityDraft** ([Long](#) *entityId*, [String](#) *username*)

getEntityFields

public [List](#)<[FieldDto](#)> **getEntityFields** ([Long](#) *entityId*)

getEntityForEdit

public [EntityDto](#) **getEntityForEdit** ([Long](#) *entityId*)

getEntityLookups

public [List](#)<[LookupDto](#)> **getEntityLookups** ([Long](#) *entityId*)

getFields

```
public List<FieldDto> getFields (Long entityId)
```

getLookupByName

```
public LookupDto getLookupByName (Long entityId, String lookupName)
```

listEntities

```
public List<EntityDto> listEntities ()
```

listEntities

```
public List<EntityDto> listEntities (boolean withSecurityCheck)
```

listWorkInProgress

```
public List<EntityDto> listWorkInProgress ()
```

saveDraftEntityChanges

```
public DraftResult saveDraftEntityChanges (Long entityId, DraftData draftData, String username)
```

saveDraftEntityChanges

```
public DraftResult saveDraftEntityChanges (Long entityId, DraftData draftData)
```

setAllEntities

```
public void setAllEntities (AllEntities allEntities)
```

setAllEntityAudits

```
public void setAllEntityAudits (AllEntityAudits allEntityAudits)
```

setAllEntityDrafts

```
public void setAllEntityDrafts (AllEntityDrafts allEntityDrafts)
```

setAllTypes

```
public void setAllTypes (AllTypes allTypes)
```

setMDSConstructor

```
public void setMDSConstructor (MDSConstructor mdsConstructor)
```

updateComboboxValues

```
public void updateComboboxValues (Long entityId, Map<String, Collection> fieldValuesToUpdate)
```

updateDraft

```
public EntityDto updateDraft (Long entityId)
```

7.49.3 HistoryServiceImpl

```
public class HistoryServiceImpl extends BasePersistenceService implements HistoryService  
    Default implementation of org.motechproject.mds.service.HistoryService interface.
```

Methods**countHistoryRecords**

```
public long countHistoryRecords (Object instance)
```

create

```
protected <T> Object create (Class<T> clazz, Object src, EntityType type)
```

getHistoryForInstance

```
public List getHistoryForInstance (Object instance, QueryParams queryParams)
```

record

```
public void record (Object instance)
```

remove

```
public void remove (Object instance)
```

setTrashFlag

```
public void setTrashFlag (Object instance, Object trash, boolean flag)
```

7.49.4 JarGeneratorServiceImpl

public class **JarGeneratorServiceImpl** implements **JarGeneratorService**

Default implementation of `org.motechproject.mds.service.JarGeneratorService` interface.

Methods

generate

public `File` **generate** ()

regenerateMdsDataBundle

public synchronized void **regenerateMdsDataBundle** (boolean *buildDDE*)

regenerateMdsDataBundle

public void **regenerateMdsDataBundle** (boolean *buildDDE*, boolean *startBundle*)

regenerateMdsDataBundleAfterDdeEnhancement

public void **regenerateMdsDataBundleAfterDdeEnhancement** (`String` *moduleName*)

setBundleContext

public void **setBundleContext** (`BundleContext` *bundleContext*)

setMdsConstructor

public void **setMdsConstructor** (`MDSConstruktor` *mdsConstructor*)

setMdsDataProvider

public void **setMdsDataProvider** (`MDSDDataProvider` *mdsDataProvider*)

setMetadataHolder

public void **setMetadataHolder** (`MetadataHolder` *metadataHolder*)

setMonitor

public void **setMonitor** (`EntitiesBundleMonitor` *monitor*)

setVelocityEngine

public void **setVelocityEngine** (VelocityEngine *velocityEngine*)

7.49.5 MdsScheduledJob

public class **MdsScheduledJob** implements Job
Job responsible for emptying MDS trash.

Methods

execute

public void **execute** (JobExecutionContext *jobExecutionContext*)

7.49.6 MdsSchedulerServiceImpl

public class **MdsSchedulerServiceImpl** implements [MdsSchedulerService](#)

Fields

DEFAULT_WAIT_TIME

public static final int **DEFAULT_WAIT_TIME**

JOB_GROUP_NAME

public static final [String](#) **JOB_GROUP_NAME**

MAX_REPEAT_COUNT

public static final int **MAX_REPEAT_COUNT**

RETRIEVAL_RETRIES_COUNT

public static final int **RETRIEVAL_RETRIES_COUNT**

SCHEDULER_SYMBOLIC_NAME

public static final [String](#) **SCHEDULER_SYMBOLIC_NAME**

Constructors

MdsSchedulerServiceImpl

public **MdsSchedulerServiceImpl** ([BundleContext](#) *bundleContext*)

Methods

scheduleRepeatingJob

public void **scheduleRepeatingJob** (long *interval*)

unscheduleRepeatingJob

public void **unscheduleRepeatingJob** ()

7.49.7 TrashServiceImpl

public class **TrashServiceImpl** extends [BasePersistenceService](#) implements [TrashService](#)
Default implementation of [org.motechproject.mds.service.TrashService](#) interface.

Methods

countTrashRecords

public long **countTrashRecords** ([String](#) *className*)

emptyTrash

public void **emptyTrash** ()

findTrashById

public [Object](#) **findTrashById** ([Object](#) *instanceId*, [Object](#) *entityId*)

getInstancesFromTrash

public [Collection](#) **getInstancesFromTrash** ([String](#) *className*, [QueryParams](#) *queryParams*)

isTrashMode

public boolean **isTrashMode** ()

moveFromTrash

```
public void moveFromTrash (Object newInstance, Object trash)
```

moveToTrash

```
public void moveToTrash (Object instance, Long entityVersion)
```

scheduleEmptyTrashJob

```
public void scheduleEmptyTrashJob ()
```

setHistoryService

```
public void setHistoryService (HistoryService historyService)
```

setMdsSchedulerService

```
public void setMdsSchedulerService (MdsSchedulerService mdsSchedulerService)
```

setSettingsService

```
public void setSettingsService (SettingsService settingsService)
```

7.49.8 TypeServiceImpl

```
public class TypeServiceImpl implements TypeService
```

```
    Default implementation of org.motechproject.mds.service.TypeService interface
```

Methods**findType**

```
public TypeDto findType (Class<?> clazz)
```

findValidations

```
public List<TypeValidation> findValidations (TypeDto type, Class<? extends Annotation> aClass)
```

getAllTypes

```
public List<TypeDto> getAllTypes ()
```

getType

```
public Type getType (TypeValidation validation)
```

setAllTypeValidations

```
public void setAllTypeValidations (AllTypeValidations allTypeValidations)
```

setAllTypes

```
public void setAllTypes (AllTypes allTypes)
```

7.50 org.motechproject.mds.util

7.50.1 ClassName

```
public final class ClassName
```

The `ClassName` util provides several methods which should help for example with getting class name or package from string representation of class. There is also methods related with creating names for repository, service interface and implementation of this service interface.

Methods

getEntityName

```
public static String getEntityName (String className)
```

getHistoryClassName

```
public static String getHistoryClassName (String className)
```

getInterfaceName

```
public static String getInterfaceName (String className)
```

getPackage

```
public static String getPackage (String className)
```

getRepositoryName

```
public static String getRepositoryName (String className)
```

getServiceName

```
public static String getServiceName (String className)
```

getSimpleName

```
public static String getSimpleName (String className)
```

getTrashClassName

```
public static String getTrashClassName (String className)
```

trimTrashHistorySuffix

```
public static String trimTrashHistorySuffix (String className)
```

7.50.2 Constants

```
public final class Constants
```

The `Constants` contains constant values used in MDS module. They are grouped by their role.

7.50.3 Constants.AnnotationFields

```
public static final class AnnotationFields
```

The `AnnotationFields` contains constant values related with attributes names in mds annotations.

See also: `org.motechproject.mds.annotations.Entity`, `org.motechproject.mds.annotations.Field`, `org.motechproject.mds.annotations.Ignore`, `org.motechproject.mds.annotations.Lookup`, `org.motechproject.mds.annotations.LookupField`

Fields

DELETE

```
public static final String DELETE
```

Constant `DELETE` corresponding to the attribute name `delete`

DISPLAY_NAME

```
public static final String DISPLAY_NAME
```

Constant `DISPLAY_NAME` corresponding to the primitive value `displayName`

FRACTION

```
public static final String FRACTION
```

Constant `FRACTION` corresponding to the primitive value `fraction`

INTEGER

public static final [String](#) **INTEGER**

Constant INTEGER corresponding to the primitive value `integer`

MAX

public static final [String](#) **MAX**

Constant MAX corresponding to the primitive value `max`

MIN

public static final [String](#) **MIN**

Constant MIN corresponding to the primitive value `min`

MODULE

public static final [String](#) **MODULE**

Constant MODULE corresponding to the attribute name `module`

NAME

public static final [String](#) **NAME**

Constant NAME corresponding to the attribute name `name`

NAMESPACE

public static final [String](#) **NAMESPACE**

Constant NAMESPACE corresponding to the attribute name `namespace`

PERSIST

public static final [String](#) **PERSIST**

Constant PERSIST corresponding to the attribute name `persist`

REGEXP

public static final [String](#) **REGEXP**

Constant REGEXP corresponding to the primitive value `regexp`

UPDATE

public static final [String](#) **UPDATE**

Constant UPDATE corresponding to the attribute name `update`

VALUE

public static final [String](#) **VALUE**
Constant **VALUE** corresponding to the primitive value `value`

7.50.4 Constants.BundleNames

public static final class **BundleNames**
The names of the mds bundles.

Fields

MDS_BUNDLE_NAME

public static final [String](#) **MDS_BUNDLE_NAME**

MDS_BUNDLE_SYMBOLIC_NAME

public static final [String](#) **MDS_BUNDLE_SYMBOLIC_NAME**

MDS_ENTITIES_NAME

public static final [String](#) **MDS_ENTITIES_NAME**

MDS_ENTITIES_SYMBOLIC_NAME

public static final [String](#) **MDS_ENTITIES_SYMBOLIC_NAME**

MDS_MIGRATION_NAME

public static final [String](#) **MDS_MIGRATION_NAME**

MDS_MIGRATION_SYMBOLIC_NAME

public static final [String](#) **MDS_MIGRATION_SYMBOLIC_NAME**

SYMBOLIC_NAME_PREFIX

public static final [String](#) **SYMBOLIC_NAME_PREFIX**

7.50.5 Constants.Config

public static final class **Config**

The `Config` contains constant values related with properties inside files:

- `datanucleus.properties`
- `motech-mds.properties`

Fields

DATANUCLEUS_FILE

public static final `String` **DATANUCLEUS_FILE**

Constant `DATANUCLEUS_FILE` presents the file name with configuration for datanucleus.

EMPTY_TRASH_JOB

public static final `String` **EMPTY_TRASH_JOB**

Constant `EMPTY_TRASH_JOB` presents a name of job scheduled by scheduler module.

MDS_DELETE_MODE

public static final `String` **MDS_DELETE_MODE**

Constant `MDS_DELETE_MODE` presents what should happen with objects when there are deleted. They can be deleted permanently or moved to the trash. The following values are valid for this property:

- `delete`
- `trash`

MDS_EMPTY_TRASH

public static final `String` **MDS_EMPTY_TRASH**

The boolean property that specifies if the trash should be empty after some time.

See also: `.MDS_DELETE_MODE`, `.MDS_TIME_VALUE`, `.MDS_TIME_UNIT`

MDS_TIME_UNIT

public static final `String` **MDS_TIME_UNIT**

The property that specifies what time unit should be used to specify time when trash should be cleaned. The following values are valid for this property:

- `Hours`
- `Days`
- `Weeks`
- `Months`
- `Years`

See also: `.MDS_DELETE_MODE`, `.MDS_EMPTY_TRASH`, `.MDS_TIME_VALUE`

MDS_TIME_VALUE

public static final [String](#) **MDS_TIME_VALUE**

The integer property that specifies after what time (according with correct time unit) trash should be cleaned.

See also: `.MDS_DELETE_MODE`, `.MDS_EMPTY_TRASH`, `.MDS_TIME_UNIT`

MODULE_FILE

public static final [String](#) **MODULE_FILE**

Constant `MODULE_FILE` presents the file name with configuration for MDS module.

7.50.6 Constants.Manifest

public static final class **Manifest**

The `Manifest` contains constant values related with attributes inside the `motech-platform-dataservices-entities` bundle manifest.

See also: `org.motechproject.mds.service.JarGeneratorService`,
`org.motechproject.mds.service.impl.JarGeneratorServiceImpl`

Fields

BUNDLE_MANIFESTVERSION

public static final [String](#) **BUNDLE_MANIFESTVERSION**

Constant `BUNDLE_MANIFESTVERSION` presents a version of bundle manifest.

BUNDLE_NAME_SUFFIX

public static final [String](#) **BUNDLE_NAME_SUFFIX**

Constant `BUNDLE_NAME_SUFFIX` presents suffix of the name of bundle that will be created by implementation of `org.motechproject.mds.service.JarGeneratorService` interface.

MANIFEST_VERSION

public static final [String](#) **MANIFEST_VERSION**

Constant `MANIFEST_VERSION` presents a version of jar manifest.

SYMBOLIC_NAME_SUFFIX

public static final [String](#) **SYMBOLIC_NAME_SUFFIX**

Constant `SYMBOLIC_NAME_SUFFIX` presents suffix of the bundle symbolic name of bundle that will be created by implementation of `org.motechproject.mds.service.JarGeneratorService` interface.

7.50.7 Constants.MetadataKeys

public static final class **MetadataKeys**
The keys used in fields metadata

Fields

ENUM_CLASS_NAME

public static final [String](#) **ENUM_CLASS_NAME**

MAP_KEY_TYPE

public static final [String](#) **MAP_KEY_TYPE**

MAP_VALUE_TYPE

public static final [String](#) **MAP_VALUE_TYPE**

RELATED_CLASS

public static final [String](#) **RELATED_CLASS**

RELATED_FIELD

public static final [String](#) **RELATED_FIELD**

7.50.8 Constants.Operators

public static final class **Operators**
Operators that users can use in lookups.

Fields

ENDS_WITH

public static final [String](#) **ENDS_WITH**

EQ

public static final [String](#) **EQ**

EQ_IGNORE_CASE

public static final [String](#) **EQ_IGNORE_CASE**

GT

public static final [String](#) **GT**

GT_EQ

public static final [String](#) **GT_EQ**

LT

public static final [String](#) **LT**

LT_EQ

public static final [String](#) **LT_EQ**

MATCHES

public static final [String](#) **MATCHES**

NEQ

public static final [String](#) **NEQ**

STARTS_WITH

public static final [String](#) **STARTS_WITH**

7.50.9 Constants.Packages

public static final class **Packages**

The `Packages` contains constant values related with packages inside MDS module.

Fields

BASE

public static final [String](#) **BASE**

Constant `BASE` presents the base package for all packages inside MDS module.

ENTITY

public static final [String](#) **ENTITY**

Constant `ENTITY` presents a package for entity classes.

See also: `.BASE`

REPOSITORY

public static final [String](#) **REPOSITORY**

Constant `REPOSITORY` presents a package for repository classes.

See also: `.BASE`

SERVICE

public static final [String](#) **SERVICE**

Constant `SERVICE` presents a package for service interfaces.

See also: `.BASE`

SERVICE_IMPL

public static final [String](#) **SERVICE_IMPL**

Constant `SERVICE_IMPL` presents a package for implementation of interfaces defined in `SERVICE` package.

See also: `.BASE`, `.SERVICE`

7.50.10 Constants.PackagesGenerated

public static final class **PackagesGenerated**

Fields

ENTITY

public static final [String](#) **ENTITY**

Constant `ENTITY` presents a package for generated entity classes.

REPOSITORY

public static final [String](#) **REPOSITORY**

Constant `REPOSITORY` presents a package for generated repository classes.

See also: `.ENTITY`

SERVICE

public static final [String](#) **SERVICE**

Constant `SERVICE` presents a package for generated service interfaces.

See also: `.ENTITY`

SERVICE_IMPL

public static final [String](#) **SERVICE_IMPL**

Constant `SERVICE_IMPL` presents a package for generated implementation of interfaces defined in [SERVICE](#) package.

See also: `.SERVICE`

7.50.11 Constants.Roles

public static final class **Roles**

The `Roles` contains constant values related with security roles.

Fields

DATA_ACCESS

public static final [String](#) **DATA_ACCESS**

Users with 'Data Access' have the ability to view the Data Browser tab. From that tab then can search for objects within the system, view and modify the data stored in the system.

HAS_ANY_MDS_ROLE

public static final [String](#) **HAS_ANY_MDS_ROLE**

Spring security el expression to check if the given user has any of the MDS roles.

See also: `.SCHEMA_ACCESS`, `.SETTINGS_ACCESS`, `.DATA_ACCESS`

HAS_DATA_ACCESS

public static final [String](#) **HAS_DATA_ACCESS**

Spring security el expression to check if the given user has the 'Data Access' role.

See also: `.DATA_ACCESS`

HAS_DATA_OR_SCHEMA_ACCESS

public static final [String](#) **HAS_DATA_OR_SCHEMA_ACCESS**

Spring security el expression to check if the given user has the 'Schema Access' or 'Data Access' roles.

See also: `.SCHEMA_ACCESS`, `.DATA_ACCESS`

HAS_SCHEMA_ACCESS

public static final [String](#) **HAS_SCHEMA_ACCESS**

Spring security el expression to check if the given user has the 'Schema Access' role.

See also: `.SCHEMA_ACCESS`

HAS_SETTINGS_ACCESS

public static final [String](#) **HAS_SETTINGS_ACCESS**

Spring security el expression to check if the given user has the 'Settings Access' role.

See also: `.SETTINGS_ACCESS`

SCHEMA_ACCESS

public static final [String](#) **SCHEMA_ACCESS**

Users with 'Schema Access' have the ability to view the Schema Editor tab of the UI. Then can add new objects, delete existing objects and modify the fields on existing objects.

SETTINGS_ACCESS

public static final [String](#) **SETTINGS_ACCESS**

Users with 'Settings Access' have the ability to view the Settings tab. From that tab then can modify data retention policies as well as import and export schema and data.

7.50.12 Constants.Settings

public static final class **Settings**

Keys for entity settings.

Fields

ALLOW_MULTIPLE_SELECTIONS

public static final [String](#) **ALLOW_MULTIPLE_SELECTIONS**

ALLOW_USER_SUPPLIED

public static final [String](#) **ALLOW_USER_SUPPLIED**

COMBOBOX_VALUES

public static final [String](#) **COMBOBOX_VALUES**

STRING_MAX_LENGTH

public static final [String](#) **STRING_MAX_LENGTH**

7.50.13 Constants.Util

public static final class **Util**

The `Util` contains constant values to help avoid string literal repetition.

See also: `pmd`

Fields

CREATION_DATE_DISPLAY_FIELD_NAME

public static final `String` **CREATION_DATE_DISPLAY_FIELD_NAME**

CREATION_DATE_FIELD_NAME

public static final `String` **CREATION_DATE_FIELD_NAME**

CREATOR_DISPLAY_FIELD_NAME

public static final `String` **CREATOR_DISPLAY_FIELD_NAME**

CREATOR_FIELD_NAME

public static final `String` **CREATOR_FIELD_NAME**

DATANUCLEUS

public static final `String` **DATANUCLEUS**

ENTITY

public static final `String` **ENTITY**

Constant **ENTITY** corresponding to the field name of the class that want to create a bidirectional connection with instane of `org.motechproject.mds.domain.Entity`

FALSE

public static final `String` **FALSE**

Constant **FALSE** corresponding to the primitive value `false`

ID_DISPLAY_FIELD_NAME

public static final `String` **ID_DISPLAY_FIELD_NAME**

ID_FIELD_NAME

public static final `String` **ID_FIELD_NAME**

MODIFICATION_DATE_DISPLAY_FIELD_NAME

public static final `String` **MODIFICATION_DATE_DISPLAY_FIELD_NAME**

MODIFICATION_DATE_FIELD_NAME

```
public static final String MODIFICATION_DATE_FIELD_NAME
```

MODIFIED_BY_DISPLAY_FIELD_NAME

```
public static final String MODIFIED_BY_DISPLAY_FIELD_NAME
```

MODIFIED_BY_FIELD_NAME

```
public static final String MODIFIED_BY_FIELD_NAME
```

OWNER_DISPLAY_FIELD_NAME

```
public static final String OWNER_DISPLAY_FIELD_NAME
```

OWNER_FIELD_NAME

```
public static final String OWNER_FIELD_NAME
```

TRUE

```
public static final String TRUE
```

Constant TRUE corresponding to the primitive value `true`

7.50.14 FieldHelper

```
public final class FieldHelper
```

Utility class handling dynamic setting of field values

Methods**setField**

```
public static void setField(Object current, String path, List value)
```

7.50.15 HistoryFieldUtil

```
public final class HistoryFieldUtil
```

The `HistoryFieldUtil` class provides helper methods to determine field names in the given history class.

Methods**currentVersion**

```
public static String currentVersion(Class<?> historyClass)
```

isLast

```
public static String isLast (Class<?> historyClass)
```

schemaVersion

```
public static String schemaVersion (Class<?> historyClass)
```

trashFlag

```
public static String trashFlag (Class<?> historyClass)
```

7.50.16 InstanceSecurityRestriction

```
public class InstanceSecurityRestriction
```

Represents a restriction on entity instances

Methods

isByCreator

```
public boolean isByCreator ()
```

isByOwner

```
public boolean isByOwner ()
```

isEmpty

```
public boolean isEmpty ()
```

setByCreator

```
public void setByCreator (boolean byCreator)
```

setByOwner

```
public void setByOwner (boolean byOwner)
```

7.50.17 Loader

```
public abstract class Loader<T>
```

The `Loader` is an abstract class that checks if all class dependencies to the given class definition are resolved. If not then the missing class name is taken from exception and the `doWhenClassNotFound(String)` method is executed.

Parameters

- `<T>` – the type of argument data

Methods**doWhenClassNotFound**

public abstract void **doWhenClassNotFound** (*String name*)

getClassDefinition

public abstract *Class*<?> **getClassDefinition** (*T arg*)

loadClass

public *Class*<?> **loadClass** (*T arg*)

7.50.18 LookupName

public final class **LookupName**

Utility class for dealing with lookup names.

Methods**lookupCountMethod**

public static *String* **lookupCountMethod** (*String lookupNameOrMethodName*)

lookupMethod

public static *String* **lookupMethod** (*String lookupName*)

7.50.19 MDSCClassLoader

public class **MDSCClassLoader** extends *ClassLoader*

The MDSCClassLoader class is a mds wrapper for *ClassLoader*.

Constructors**MDSCClassLoader**

protected **MDSCClassLoader** ()

MDSCClassLoader

protected **MDSCClassLoader** (*ClassLoader parent*)

Methods

defineClass

public *Class*<?> **defineClass** (*String name*, byte[] *bytecode*)

getInstance

public static *MDSCClassLoader* **getInstance** ()

getStandaloneInstance

public static *MDSCClassLoader* **getStandaloneInstance** ()

getStandaloneInstance

public static *MDSCClassLoader* **getStandaloneInstance** (*ClassLoader parent*)

reloadClassLoader

public static void **reloadClassLoader** ()

safeDefineClass

public *Class*<?> **safeDefineClass** (*String name*, byte[] *bytecode*)

7.50.20 MemberUtil

public final class **MemberUtil**

Fields

FIELD_NAME_START_IDX

public static final *Integer* **FIELD_NAME_START_IDX**

GETTER_PREFIX

public static final *String* **GETTER_PREFIX**

SETTER_PREFIX

```
public static final String SETTER_PREFIX
```

Methods

getCorrectType

```
public static Class<?> getCorrectType (AnnotatedElement object)
```

getCorrectType

```
public static Class<?> getCorrectType (Member object)
```

getFieldName

```
public static String getFieldName (AnnotatedElement object)
```

getFieldName

```
public static String getFieldName (Member object)
```

getGenericType

```
public static Class<?> getGenericType (AnnotatedElement object)
```

getGenericType

```
public static Class<?> getGenericType (AnnotatedElement object, int typeNumber)
```

getGenericType

```
public static Class<?> getGenericType (Member object, int typeNumber)
```

getMembers

```
public static List<Member> getMembers (Class<?> clazz, Predicate methodPredicate, Predicate fieldPredicate)
```

7.50.21 NumberPredicate

```
public class NumberPredicate implements Predicate
```

Constructors

NumberPredicate

```
public NumberPredicate (Number element)
```

Methods

evaluate

```
public boolean evaluate (Object candidate)
```

7.50.22 ObjectReference

```
public class ObjectReference
```

Represents an object reference. It holds an information about related field name, as well as the object that the field should reference to.

Constructors

ObjectReference

```
public ObjectReference (String fieldName, Object reference)
```

Methods

getFieldName

```
public String getFieldName ()
```

getReference

```
public Object getReference ()
```

setFieldName

```
public void setFieldName (String fieldName)
```

setReference

```
public void setReference (Object reference)
```

7.50.23 Order

```
public class Order implements Serializable
```

Represents an order in a query

Constructors

Order

```
public Order (String field)
```

Order

```
public Order (String field, String direction)
```

Order

```
public Order (String field, Direction direction)
```

Methods

getDirection

```
public Direction getDirection ()
```

getField

```
public String getField ()
```

toString

```
public String toString ()
```

7.50.24 Order.Direction

```
public static enum Direction
```

Enum Constants

ASC

```
public static final Order.Direction ASC
```

DESC

```
public static final Order.Direction DESC
```

7.50.25 Pair

public interface **Pair**<N, V>

The `Pair` util interface should use everywhere where developer needs a pair of key-value

Parameters

- <N> – type of key
- <V> – type of value

See also: `org.motechproject.mds.domain.FieldMetadata`, `org.motechproject.mds.domain.FieldSet`, `org.motechproject.mds.dto.MetadataDto`, `org.motechproject.mds.dto.SettingDto`

Methods

`getKey`

N `getKey()`

`getValue`

V `getValue()`

7.50.26 PropertyUtil

public final class **PropertyUtil** extends `PropertyUtils`

The `PropertyUtil` util class provides the same method like `org.apache.commons.beanutils.PropertyUtils` and two additional methods for safe writing and reading property in the given bean.

Methods

`getPropertyDescriptors`

public static `PropertyDescriptor[]` `getPropertyDescriptors` (`Object bean`)

`safeGetProperty`

public static `Object` `safeGetProperty` (`Object bean`, `String name`)

`safeSetProperty`

public static void `safeSetProperty` (`Object bean`, `String name`, `Object value`)

7.50.27 SecurityMode

public enum **SecurityMode**

This enum describes security mode for an entity

Enum Constants

CREATOR

public static final [SecurityMode](#) **CREATOR**

EVERYONE

public static final [SecurityMode](#) **EVERYONE**

OWNER

public static final [SecurityMode](#) **OWNER**

ROLES

public static final [SecurityMode](#) **ROLES**

USERS

public static final [SecurityMode](#) **USERS**

7.50.28 SecurityUtil

public final class **SecurityUtil**

The `SecurityUtil` class provides helper methods to retrieve logged user details, such as username or roles

Methods

`getUserRoles`

public static [List<String>](#) **getUserRoles** ()

`getUsername`

public static [String](#) **getUsername** ()

7.50.29 TypeHelper

public final class **TypeHelper**

A helper class for parsing and formatting mds supported types.

Methods

breakString

```
public static String[] breakString (String str)
```

breakString

```
public static String[] breakString (String str, String[] removes, String[] search, String[] replacement, String separator)
```

breakStringForList

```
public static String[] breakStringForList (String str)
```

format

```
public static String format (Object obj)
```

getPrimitive

```
public static Class<?> getPrimitive (Class<?> clazz)
```

getWrapperForPrimitive

```
public static Class<?> getWrapperForPrimitive (Class<?> clazz)
```

hasPrimitive

```
public static boolean hasPrimitive (Class<?> clazz)
```

isPrimitive

```
public static boolean isPrimitive (Class<?> clazz)
```

isPrimitive

```
public static boolean isPrimitive (String className)
```

parse

```
public static Object parse (Object val, Class<?> toClass)
```


parse

public static Object **parse** (Object *val*, String *toClass*)

parse

public static Object **parse** (Object *val*, String *toClass*, ClassLoader *classLoader*)

parse

public static Object **parse** (Object *val*, String *toClass*, String *genericType*)

parse

public static Object **parse** (Object *val*, String *toClass*, String *genericType*, ClassLoader *classLoader*)

parseIntToBool

public static boolean **parseIntToBool** (Integer *val*)

parseList

public static List **parseList** (List *val*, Class<?> *generic*)

parseNumber

public static Number **parseNumber** (Object *val*, String *toClass*)

parseString

public static Object **parseString** (String *str*, Class<?> *toClass*)

parseString

public static Object **parseString** (String *str*, String *toClass*)

parseString

public static Object **parseString** (String *str*, Class<?> *toClass*, Class<?> *generic*)

parseStringToMap

public static Map **parseStringToMap** (String *str*)

toRange

public static [Range](#) **toRange** ([Object](#) *object*, [String](#) *typeClass*)

toSet

public static [Set](#) **toSet** ([Object](#) *object*, [String](#) *typeClass*)

7.50.30 ValidationUtil

public final class **ValidationUtil**
Common validation utils for mds.

Methods

validateNoJavaKeyword

public static void **validateNoJavaKeyword** ([String](#) *str*)

7.51 org.motechproject.osgi.web

7.51.1 Activator

public class **Activator** implements [BundleActivator](#)

Methods

resourceMappings

protected [Map](#)<[String](#), [String](#)> **resourceMappings** ()

start

public void **start** ([BundleContext](#) *context*)

stop

public void **stop** ([BundleContext](#) *context*)

7.51.2 ApplicationContextTracker

public abstract class **ApplicationContextTracker** extends [ServiceTracker](#)

Base class for every class that wishes to track Spring application context. Contains a methods that help with synchronous processing.

Constructors

ApplicationContextTracker

```
public ApplicationContextTracker (BundleContext context)
```

Methods

contextInvalidOrProcessed

```
protected boolean contextInvalidOrProcessed (ServiceReference serviceReference, ApplicationCon-  
text applicationContext)
```

getLock

```
protected Object getLock ()
```

markAsProcessed

```
protected void markAsProcessed (ApplicationContext applicationContext)
```

removeFromProcessed

```
protected void removeFromProcessed (ApplicationContext applicationContext)
```

7.51.3 BlueprintActivator

```
public class BlueprintActivator implements BundleActivator
```

Methods

start

```
public void start (BundleContext context)
```

stop

```
public void stop (BundleContext context)
```

7.51.4 BlueprintApplicationContextTracker

```
public class BlueprintApplicationContextTracker extends ApplicationContextTracker
```

The `BlueprintApplicationContextTracker` class tracks application contexts, which are registered as services.

Constructors

BlueprintApplicationContextTracker

```
public BlueprintApplicationContextTracker (BundleContext context)
```

Methods

addingService

```
public Object addingService (ServiceReference serviceReference)
```

removedService

```
public void removedService (ServiceReference reference, Object service)
```

7.51.5 BundleContextWrapper

```
public class BundleContextWrapper implements BundleContextAware
```

Fields

CONTEXT_SERVICE_NAME

```
public static final String CONTEXT_SERVICE_NAME
```

Constructors

BundleContextWrapper

```
public BundleContextWrapper ()
```

BundleContextWrapper

```
public BundleContextWrapper (BundleContext context)
```

Methods

getBundleApplicationContext

```
public ApplicationContext getBundleApplicationContext ()
```

getBundleContext

```
public BundleContext getBundleContext ()
```

getCurrentBundleSymbolicName

```
public String getCurrentBundleSymbolicName ()
```

getService

```
public <T> T getService (Class<T> clazz)
```

setBundleContext

```
public void setBundleContext (BundleContext bundleContext)
```

7.51.6 BundleRegister

```
public final class BundleRegister
```

The `BundleRegister` Singleton class is used for recording bundles. This class will help to reconfigure logger's levels.

Methods**addBundle**

```
public void addBundle (Bundle bundle)
```

getBundleList

```
public List<Bundle> getBundleList ()
```

getInstance

```
public static BundleRegister getInstance ()
```

7.51.7 BundledJspView

```
public class BundledJspView extends JstlView implements BundleContextAware
```

Methods**render**

```
public void render (Map<String, ?> model, HttpServletRequest request, HttpServletResponse response)
```

setBundleContext

```
public void setBundleContext (BundleContext bundleContext)
```

7.51.8 `HttpServiceTracker`

public class **HttpServiceTracker** extends `ServiceTracker`

Constructors

`HttpServiceTracker`

public **HttpServiceTracker** (`BundleContext` *context*, `Map`<`String`, `String`> *resourceMapping*)

Methods

`addingService`

public `Object` **addingService** (`ServiceReference` *serviceReference*)

`removedService`

public void **removedService** (`ServiceReference` *ref*, `Object` *service*)

`start`

public void **start** ()

`unregister`

public void **unregister** ()

7.51.9 `HttpServiceTrackers`

public class **HttpServiceTrackers**

Methods

`addTrackerFor`

public `HttpServiceTracker` **addTrackerFor** (`Bundle` *bundle*)

`isBeingTracked`

public boolean **isBeingTracked** (`Bundle` *bundle*)

`removeTrackerFor`

public `HttpServiceTracker` **removeTrackerFor** (`Bundle` *bundle*)

7.51.10 Log4JBundleLoader

public class **Log4JBundleLoader**

This `Log4JBundleLoader` class is responsible for loading configuration of the loggers from the saved properties or file in bundle (`log4j.xml`).

Methods

`checkListContainLogger`

public boolean **checkListContainLogger** (`List<LogMapping>` *loggers*, `String` *log*)

`checkLogXmlConfiguration`

public boolean **checkLogXmlConfiguration** (`Document` *log4jDoc*)

`createLoggerProperties`

public `Properties` **createLoggerProperties** (`List<LogMapping>` *log*)

`loadBundle`

public void **loadBundle** (`Bundle` *bundle*)

`loadLoggerDbConfiguration`

public void **loadLoggerDbConfiguration** ()

`setLog4jConf`

public void **setLog4jConf** (`String` *log4jConf*)

7.51.11 ModuleRegistrationData

public class **ModuleRegistrationData**

Object used to registered a module withing the Motech UI system. Represents a module and is used for building the common user interface. All modules that wish to register within the UI system must either expose this class as a spring bean in their application context or manually register it through the `UIFrameworkService` OSGi service.

See also: `UIFrameworkService`

Constructors

`ModuleRegistrationData`

public **ModuleRegistrationData** ()

ModuleRegistrationData

```
public ModuleRegistrationData (String moduleName, String url)
```

ModuleRegistrationData

```
public ModuleRegistrationData (String moduleName, Map<String, String> i18n)
```

ModuleRegistrationData

```
public ModuleRegistrationData (String moduleName, String url, List<String> angularModules,  
                                Map<String, String> i18n)
```

Methods

addAngularModule

```
public void addAngularModule (String moduleName)
```

addI18N

```
public void addI18N (String fileName, String fileLocation)
```

addSubMenu

```
public void addSubMenu (String url, String label)
```

addSubMenu

```
public void addSubMenu (String url, String label, String roleForAccess)
```

equals

```
public boolean equals (Object o)
```

getAngularModules

```
public List<String> getAngularModules ()
```

getAngularModulesStr

```
public String getAngularModulesStr ()
```


getBundle

```
public Bundle getBundle ()
```

getCriticalMessage

```
public String getCriticalMessage ()
```

getDefaultURL

```
public String getDefaultURL ()
```

getI18n

```
public Map<String, String> getI18n ()
```

getModuleName

```
public String getModuleName ()
```

getResourcePath

```
public String getResourcePath ()
```

getRoleForAccess

```
public List<String> getRoleForAccess ()
```

getSettingsURL

```
public String getSettingsURL ()
```

getSubMenu

```
public Map<String, SubmenuInfo> getSubMenu ()
```

getUrl

```
public String getUrl ()
```

hashCode

```
public int hashCode ()
```

isNeedsAttention

public boolean **isNeedsAttention** ()

removeAngularModule

public void **removeAngularModule** (*String moduleName*)

setBundle

public void **setBundle** (*Bundle bundle*)

setCriticalMessage

public void **setCriticalMessage** (*String criticalMessage*)

setDefaultURL

public void **setDefaultURL** (*String defaultURL*)

setModuleName

public void **setModuleName** (*String moduleName*)

setNeedsAttention

public void **setNeedsAttention** (boolean *needsAttention*)

setResourcePath

public void **setResourcePath** (*String resourcePath*)

setRoleForAccess

public void **setRoleForAccess** (*String role*)

setRoleForAccess

public void **setRoleForAccess** (*List<String> roles*)

setSettingsURL

public void **setSettingsURL** (*String settingsURL*)

setSubMenu

```
public void setSubMenu (Map<String, SubmenuInfo> subMenu)
```

setUrl

```
public void setUrl (String url)
```

submenuNeedsAttention

```
public void submenuNeedsAttention (String submenu)
```

submenuBackToNormal

```
public void submenuBackToNormal (String submenu)
```

7.51.12 MotechOsgiWebApplicationContext

```
public class MotechOsgiWebApplicationContext extends OsgiBundleXmlApplicationContext implements ConfigurableWebA
```

Constructors**MotechOsgiWebApplicationContext**

```
public MotechOsgiWebApplicationContext ()
```

Methods**getNamespace**

```
public String getNamespace ()
```

getServletConfig

```
public ServletConfig getServletConfig ()
```

getServletContext

```
public ServletContext getServletContext ()
```

isInitialized

```
public boolean isInitialized ()
```

setConfigLocation

public void **setConfigLocation** (*String configLocation*)

setNamespace

public void **setNamespace** (*String namespace*)

setServletConfig

public void **setServletConfig** (*ServletConfig servletConfig*)

setServletContext

public void **setServletContext** (*ServletContext servletContext*)

waitForContext

public void **waitForContext** (*int waitTimeInMillis*)

7.51.13 OsgiDispatcherServlet

public class **OsgiDispatcherServlet** extends *DispatcherServlet*

Constructors

OsgiDispatcherServlet

public **OsgiDispatcherServlet** (*BundleContext bundleContext*)

OsgiDispatcherServlet

public **OsgiDispatcherServlet** (*BundleContext bundleContext*, *ConfigurableWebApplicationContext configurableWebApplicationContext*)

Methods

initFrameworkServlet

protected void **initFrameworkServlet** ()

postProcessWebApplicationContext

protected void **postProcessWebApplicationContext** (*ConfigurableWebApplicationContext wac*)

7.51.14 OsgiWebApplicationContext

public class **OsgiWebApplicationContext** implements [WebApplicationContext](#)

Constructors

OsgiWebApplicationContext

public **OsgiWebApplicationContext** ([ApplicationContext](#) *applicationContext*, [ConfigurableWebApplicationContext](#) *configurableWebApplicationContext*)

Methods

containsBean

public boolean **containsBean** ([String](#) *name*)

containsBeanDefinition

public boolean **containsBeanDefinition** ([String](#) *beanName*)

containsLocalBean

public boolean **containsLocalBean** ([String](#) *name*)

findAnnotationOnBean

public <A extends Annotation> A **findAnnotationOnBean** ([String](#) *beanName*, [Class](#)<A> *annotationType*)

getAliases

public [String](#)[] **getAliases** ([String](#) *name*)

getAutowireCapableBeanFactory

public [AutowireCapableBeanFactory](#) **getAutowireCapableBeanFactory** ()

getBean

public [Object](#) **getBean** ([String](#) *name*)

getBean

public <T> T **getBean** ([String](#) *name*, [Class](#)<T> *requiredType*)

getBean

```
public <T> T getBean (Class<T> requiredType)
```

getBean

```
public Object getBean (String name, Object... args)
```

getBeanDefinitionCount

```
public int getBeanDefinitionCount ()
```

getBeanDefinitionNames

```
public String[] getBeanDefinitionNames ()
```

getBeanNamesForType

```
public String[] getBeanNamesForType (Class<?> type)
```

getBeanNamesForType

```
public String[] getBeanNamesForType (Class<?> type, boolean includeNonSingletons, boolean allowEagerInit)
```

getBeansOfType

```
public <T> Map<String, T> getBeansOfType (Class<T> type)
```

getBeansOfType

```
public <T> Map<String, T> getBeansOfType (Class<T> type, boolean includeNonSingletons, boolean allowEagerInit)
```

getBeansWithAnnotation

```
public Map<String, Object> getBeansWithAnnotation (Class<? extends Annotation> annotationType)
```

getClassLoader

```
public ClassLoader getClassLoader ()
```

getDisplayName

```
public String getDisplayName ()
```

getEnvironment

```
public ConfigurableEnvironment getEnvironment ()
```

getId

```
public String getId ()
```

getMessage

```
public String getMessage (String code, Object[] args, String defaultMessage, Locale locale)
```

getMessage

```
public String getMessage (String code, Object[] args, Locale locale)
```

getMessage

```
public String getMessage (MessageSourceResolvable resolvable, Locale locale)
```

getParent

```
public ApplicationContext getParent ()
```

getParentBeanFactory

```
public BeanFactory getParentBeanFactory ()
```

getResource

```
public Resource getResource (String location)
```

getResources

```
public Resource[] getResources (String locationPattern)
```

getServletContext

```
public ServletContext getServletContext ()
```

getStartupDate

```
public long getStartupDate ()
```

getType

public **Class**<?> **getType** (*String name*)

isPrototype

public boolean **isPrototype** (*String name*)

isSingleton

public boolean **isSingleton** (*String name*)

isTypeMatch

public boolean **isTypeMatch** (*String name*, *Class*<?> *targetType*)

publishEvent

public void **publishEvent** (*ApplicationEvent event*)

7.51.15 SubmenuInfo

public class **SubmenuInfo**

- Class to encapsulate information about submenu Links to be shown on UI

Constructors

SubmenuInfo

public **SubmenuInfo** (*String url*)

Methods

getCriticalMessage

public *String* **getCriticalMessage** ()

getRoleForAccess

public *List*<*String*> **getRoleForAccess** ()

getUrl

public *String* **getUrl** ()

isNeedsAttention

```
public boolean isNeedsAttention ()
```

setCriticalMessage

```
public void setCriticalMessage (String criticalMessage)
```

setNeedsAttention

```
public void setNeedsAttention (boolean needsAttention)
```

setRoleForAccess

```
public void setRoleForAccess (String roleForAccess)
```

setRoleForAccess

```
public void setRoleForAccess (List<String> roleForAccess)
```

setUrl

```
public void setUrl (String url)
```

7.51.16 UIFrameworkService

```
public interface UIFrameworkService
```

Service responsible for managing the interface. Provides methods for registering/un-registering modules. All modules are represented by `ModuleRegistrationData` objects, either registered directly through this service or automatically by exposing it in their spring context. This service also allows manipulation of module state, by marking given modules as requiring attention on the UI.

Fields**MODULES_WITHOUT_SUBMENU**

```
String MODULES_WITHOUT_SUBMENU
```

MODULES_WITHOUT_UI

```
String MODULES_WITHOUT_UI
```

MODULES_WITH_SUBMENU

```
String MODULES_WITH_SUBMENU
```

Methods

getModuleData

ModuleRegistrationData **getModuleData** (*String moduleName*)

getModuleDataByAngular

ModuleRegistrationData **getModuleDataByAngular** (*String angularModule*)

getModuleDataByBundle

ModuleRegistrationData **getModuleDataByBundle** (*Bundle bundle*)

getRegisteredModules

Map<String, Collection<ModuleRegistrationData>> **getRegisteredModules** ()

isModuleRegistered

boolean **isModuleRegistered** (*String moduleName*)

moduleBackToNormal

void **moduleBackToNormal** (*String moduleName*)

moduleBackToNormal

void **moduleBackToNormal** (*String moduleName, String submenu*)

moduleNeedsAttention

void **moduleNeedsAttention** (*String moduleName, String message*)

moduleNeedsAttention

void **moduleNeedsAttention** (*String moduleName, String submenu, String message*)

registerModule

void **registerModule** (*ModuleRegistrationData module*)

unregisterModule

void **unregisterModule** (*String moduleName*)

7.51.17 UIServiceTracker

public class **UIServiceTracker** extends [ServiceTracker](#)

Constructors

UIServiceTracker

public **UIServiceTracker** ([BundleContext](#) *context*, [ModuleRegistrationData](#) *moduleRegistrationData*)

UIServiceTracker

public **UIServiceTracker** ([BundleContextWrapper](#) *wrapper*, [ModuleRegistrationData](#) *moduleRegistrationData*)

Methods

addingService

public [Object](#) **addingService** ([ServiceReference](#) *ref*)

removedService

public void **removedService** ([ServiceReference](#) *ref*, [Object](#) *service*)

start

public void **start** ()

7.51.18 UIServiceTrackers

public class **UIServiceTrackers**

Methods

addTrackerFor

public [UIServiceTracker](#) **addTrackerFor** ([Bundle](#) *bundle*, [ApplicationContext](#) *applicationContext*)

isBeingTracked

public boolean **isBeingTracked** ([Bundle](#) *bundle*)

removeTrackerFor

public [UIServiceTracker](#) **removeTrackerFor** ([Bundle](#) *bundle*)

7.51.19 WebUIBundleActivator

public class **WebUIBundleActivator** extends org.motechproject.osgi.web.Activator

Methods

moduleId

protected *String* **moduleId** ()

resourceFolder

protected *String* **resourceFolder** ()

resourceMappings

protected *Map*<*String*, *String*> **resourceMappings** ()

start

public void **start** (*BundleContext context*)

stop

public void **stop** (*BundleContext context*)

7.52 org.motechproject.osgi.web.domain

7.52.1 LogMapping

public class **LogMapping**

Constructors

LogMapping

public **LogMapping** ()

LogMapping

public **LogMapping** (*String logName*, *String logLevel*)

Methods

`equals`

public boolean **equals** (*Object obj*)

`getLogLevel`

public *String* **getLogLevel** ()

`getLogName`

public *String* **getLogName** ()

`hashCode`

public int **hashCode** ()

`setLogLevel`

public void **setLogLevel** (*String logLevel*)

`setLogName`

public void **setLogName** (*String logName*)

`toString`

public *String* **toString** ()

7.53 org.motechproject.osgi.web.exception

7.53.1 RenderException

public class **RenderException** extends *Exception*

Constructors

RenderException

public **RenderException** (*String message*)

RenderException

public **RenderException** (*String message*, *Throwable cause*)

RenderException

public **RenderException** (*Throwable cause*)

7.53.2 ServletRegistrationException

public class **ServletRegistrationException** extends *RuntimeException*

Constructors

ServletRegistrationException

public **ServletRegistrationException** (*String message*)

ServletRegistrationException

public **ServletRegistrationException** (*String message*, *Throwable cause*)

ServletRegistrationException

public **ServletRegistrationException** (*Throwable cause*)

7.54 org.motechproject.osgi.web.ext

7.54.1 ApplicationEnvironment

public final class **ApplicationEnvironment**

Fields

DEVELOPMENT

public static final *String* **DEVELOPMENT**

ENVIRONMENT

public static final *String* **ENVIRONMENT**

Methods

getEnvironment

public static [String](#) **getEnvironment** ()

getModulePath

public static [String](#) **getModulePath** ([BundleName](#) *bundleName*)

isInDevelopmentMode

public static boolean **isInDevelopmentMode** ()

7.54.2 BundleName

public class **BundleName**

Constructors

BundleName

public **BundleName** ([String](#) *bundleSymbolicName*)

Methods

equals

public boolean **equals** ([Object](#) *bundle*)

hashCode

public int **hashCode** ()

underscore

public [String](#) **underscore** ()

7.54.3 FileSystemAwareUIHttpContext

public class **FileSystemAwareUIHttpContext** extends [UiHttpContext](#)

Constructors

FileSystemAwareUIHttpContext

public **FileSystemAwareUIHttpContext** ([HttpContext](#) *context*, [String](#) *resourceRootDirectoryPath*)

Methods

getResource

public [URL](#) **getResource** ([String](#) *name*)

getResourceRootDirectoryPath

public [String](#) **getResourceRootDirectoryPath** ()

7.54.4 HttpContextFactory

public final class **HttpContextFactory**

Methods

getHttpContext

public static [HttpContext](#) **getHttpContext** ([HttpContext](#) *httpContext*, [Bundle](#) *bundle*)

7.54.5 UiHttpContext

public class **UiHttpContext** implements [HttpContext](#)

Constructors

UiHttpContext

public **UiHttpContext** ([HttpContext](#) *context*)

Methods

getContext

protected [HttpContext](#) **getContext** ()

getMimeType

public [String](#) **getMimeType** ([String](#) *name*)

getResource

public URL **getResource** (*String name*)

handleSecurity

public boolean **handleSecurity** (*HttpServletRequest request, HttpServletResponse response*)

7.55 org.motechproject.osgi.web.service

7.55.1 ServerLogService

public interface **ServerLogService**

Interface for accessing Logger's configuration from the saved properties

Fields

ROOT_LOGGER_NAME

String **ROOT_LOGGER_NAME**

Methods**changeLogLevel**

void **changeLogLevel** (*String name, String level*)

changeRootLogLevel

void **changeRootLogLevel** (*String level*)

getAllLogMappings

List<LogMapping> **getAllLogMappings** ()

getLogLevels

List<LogMapping> **getLogLevels** ()

getRootLogLevel

LogMapping **getRootLogLevel** ()

reconfigure

```
void reconfigure ()
```

removeLogger

```
void removeLogger (String name)
```

7.56 org.motechproject.osgi.web.settings

7.56.1 Loggers

```
public class Loggers
```

Loggers class that holds information about all loggers in our system.

Constructors**Loggers**

```
public Loggers ()
```

Loggers

```
public Loggers (List<LogMapping> loggers, LogMapping root)
```

Methods**equals**

```
public boolean equals (Object obj)
```

getLoggers

```
public List<LogMapping> getLoggers ()
```

getRoot

```
public LogMapping getRoot ()
```

getTrash

```
public List<LogMapping> getTrash ()
```

hashCode

```
public int hashCode ()
```

setLoggers

```
public void setLoggers (List<LogMapping> loggers)
```

setRoot

```
public void setRoot (LogMapping root)
```

setTrash

```
public void setTrash (List<LogMapping> trash)
```

7.57 org.motechproject.osgi.web.util

7.57.1 BundleHeaders

```
public class BundleHeaders
```

Constructors**BundleHeaders**

```
public BundleHeaders (BundleContext bundleContext)
```

BundleHeaders

```
public BundleHeaders (Bundle bundle)
```

Methods**get**

```
public Object get (Object key)
```

getContextPath

```
public String getContextPath ()
```

getName

```
public String getName ()
```

getResourcePath

```
public String getResourcePath ()
```

getStringValue

```
public String getStringValue (String key)
```

getSymbolicName

```
public String getSymbolicName ()
```

getVersion

```
public String getVersion ()
```

isBlueprintEnabled

```
public boolean isBlueprintEnabled ()
```

7.57.2 WebBundleUtil

```
public final class WebBundleUtil
```

Utility class that's purpose is easing bundle related operations/searches.

Methods

findBundleByName

```
public static Bundle findBundleByName (BundleContext bundleContext, String name)
```

findBundleBySymbolicName

```
public static Bundle findBundleBySymbolicName (BundleContext bundleContext, String symbolic-  
Name)
```

getContextLocation

```
public static String getContextLocation (Bundle bundle)
```

getContextPath

```
public static String getContextPath (Bundle bundle)
```

getModuleId

```
public static String getModuleId (Bundle bundle)
```

7.58 org.motechproject.scheduler.builder

7.58.1 CronJobExpressionBuilder

```
public class CronJobExpressionBuilder
```

Constructors**CronJobExpressionBuilder**

```
public CronJobExpressionBuilder (Time startTime, Integer repeatWindowInHours, Integer repeatIntervalInMinutes)
```

Methods**build**

```
public String build ()
```

7.58.2 CronJobSimpleExpressionBuilder

```
public class CronJobSimpleExpressionBuilder
```

Constructors**CronJobSimpleExpressionBuilder**

```
public CronJobSimpleExpressionBuilder (Time startTime)
```

Methods**build**

```
public String build ()
```

withRepeatIntervalInDays

public [CronJobSimpleExpressionBuilder](#) **withRepeatIntervalInDays** (int *repeatIntervalInDays*)

7.58.3 WeeklyCronJobExpressionBuilder

public class **WeeklyCronJobExpressionBuilder**

Constructors

WeeklyCronJobExpressionBuilder

public **WeeklyCronJobExpressionBuilder** ([DayOfWeek](#) *dayOfWeek*)

WeeklyCronJobExpressionBuilder

public **WeeklyCronJobExpressionBuilder** (int *dayOfWeekNumber*)

Methods

build

public [String](#) **build** ()

withTime

public [WeeklyCronJobExpressionBuilder](#) **withTime** ([Time](#) *time*)

7.59 org.motechproject.scheduler.contract

7.59.1 CronJobId

public class **CronJobId** extends [JobId](#)

Constructors

CronJobId

public **CronJobId** ([String](#) *subject*, [String](#) *id*)

CronJobId

public **CronJobId** ([MotechEvent](#) *event*)

7.59.2 CronSchedulableJob

public class **CronSchedulableJob** implements [Serializable](#)

Schedulable Job - a data carrier class for a scheduled job that can be fired unlimited number of times as specified with the cron expression

Author Igor (iopushnyev@2paths.com) Date: 16/02/11 Time: 1:43 PM

Constructors

CronSchedulableJob

public **CronSchedulableJob** ([MotechEvent](#) motechEvent, [String](#) cronExpression, [Date](#) startTime, [Date](#) endTime)

CronSchedulableJob

public **CronSchedulableJob** ([MotechEvent](#) motechEvent, [String](#) cronExpression)

CronSchedulableJob

public **CronSchedulableJob** ([MotechEvent](#) motechEvent, [String](#) cronExpression, [Date](#) startTime, [Date](#) endTime, boolean ignorePastFiresAtStart)

Methods

equals

public boolean **equals** ([Object](#) obj)

getCronExpression

public [String](#) **getCronExpression** ()

getEndTime

public [Date](#) **getEndTime** ()

getMotechEvent

public [MotechEvent](#) **getMotechEvent** ()

getStartTime

public [Date](#) **getStartTime** ()

hashCode

```
public int hashCode ()
```

isIgnorePastFiresAtStart

```
public boolean isIgnorePastFiresAtStart ()
```

toString

```
public String toString ()
```

7.59.3 DayOfWeekSchedulableJob

public final class **DayOfWeekSchedulableJob** implements `Serializable`
Job that is scheduled on particular days of week

Constructors**DayOfWeekSchedulableJob**

```
public DayOfWeekSchedulableJob (MotechEvent motechEvent, LocalDate start, LocalDate end,  
                                List<DayOfWeek> days, Time time, boolean ignorePastFiresAt-  
                                Start)
```

DayOfWeekSchedulableJob

```
public DayOfWeekSchedulableJob (MotechEvent motechEvent, LocalDate start, LocalDate end,  
                                List<DayOfWeek> days, Time time)
```

Methods**getCronDays**

```
public List<Integer> getCronDays ()
```

getEndDate

```
public LocalDate getEndDate ()
```

getMotechEvent

```
public MotechEvent getMotechEvent ()
```


getStartDate

```
public LocalDate getStartDate ()
```

getTime

```
public Time getTime ()
```

isIgnorePastFiresAtStart

```
public boolean isIgnorePastFiresAtStart ()
```

7.59.4 EventInfo

```
public class EventInfo
```

EventInfo is the class which contains information about event associated with scheduled job.

Constructors

EventInfo

```
public EventInfo ()
```

Methods

getParameters

```
public Map<String, Object> getParameters ()
```

getSubject

```
public String getSubject ()
```

setParameters

```
public void setParameters (Map<String, Object> parameters)
```

setSubject

```
public void setSubject (String subject)
```

7.59.5 JobBasicInfo

```
public class JobBasicInfo
```

JobBasicInfo is the class which contains information about scheduled job and its current state.

Fields

ACTIVITY_ACTIVE

public static final [String](#) **ACTIVITY_ACTIVE**

ACTIVITY_FINISHED

public static final [String](#) **ACTIVITY_FINISHED**

ACTIVITY_NOTSTARTED

public static final [String](#) **ACTIVITY_NOTSTARTED**

JOBTYPE_CRON

public static final [String](#) **JOBTYPE_CRON**

JOBTYPE_REPEATING

public static final [String](#) **JOBTYPE_REPEATING**

JOBTYPE_RUNONCE

public static final [String](#) **JOBTYPE_RUNONCE**

STATUS_BLOCKED

public static final [String](#) **STATUS_BLOCKED**

STATUS_ERROR

public static final [String](#) **STATUS_ERROR**

STATUS_OK

public static final [String](#) **STATUS_OK**

STATUS_PAUSED

public static final [String](#) **STATUS_PAUSED**

Constructors

JobBasicInfo

public **JobBasicInfo** ()

JobBasicInfo

public **JobBasicInfo** (String activity, String status, String name, String startDate, String nextFireDate, String endDate, String jobType, String info)

Methods

getActivity

public String **getActivity** ()

getEndDate

public String **getEndDate** ()

getInfo

public String **getInfo** ()

getJobType

public String **getJobType** ()

getName

public String **getName** ()

getNextFireDate

public String **getNextFireDate** ()

getStartDate

public String **getStartDate** ()

getStatus

public String **getStatus** ()

setActivity

public void **setActivity** ([String](#) *activity*)

setEndDate

public void **setEndDate** ([String](#) *endDate*)

setInfo

public void **setInfo** ([String](#) *info*)

setJobType

public void **setJobType** ([String](#) *jobType*)

setName

public void **setName** ([String](#) *name*)

setNextFireDate

public void **setNextFireDate** ([String](#) *nextFireDate*)

setStartDate

public void **setStartDate** ([String](#) *startDate*)

setStatus

public void **setStatus** ([String](#) *status*)

7.59.6 JobDetailedInfo

public class **JobDetailedInfo**

JobDetailedInfo is the class which wraps the EventInfo list.

See also: [EventInfo](#)

Constructors

JobDetailedInfo

public **JobDetailedInfo** ()

JobDetailedInfo

public **JobDetailedInfo** ([List](#)<[EventInfo](#)> *eventInfoList*)

Methods

getEventInfoList

public [List](#)<[EventInfo](#)> **getEventInfoList** ()

setEventInfoList

public void **setEventInfoList** ([List](#)<[EventInfo](#)> *eventInfoList*)

7.59.7 JobId

public abstract class **JobId** implements [Serializable](#)

Constructors

JobId

public **JobId** ([String](#) *subject*, [String](#) *id*, [String](#) *suffix*)

JobId

public **JobId** ([MotechEvent](#) *motechEvent*, [String](#) *suffix*)

Methods

toString

public [String](#) **toString** ()

value

public [String](#) **value** ()

7.59.8 RepeatingJobId

public class **RepeatingJobId** extends [JobId](#)

Fields

SUFFIX_REPEATJOBID

public static final [String](#) SUFFIX_REPEATJOBID

Constructors

RepeatingJobId

public [RepeatingJobId](#) ([String](#) subject, [String](#) id)

RepeatingJobId

public [RepeatingJobId](#) ([MotechEvent](#) repeatingEvent)

7.59.9 RepeatingSchedulableJob

public class [RepeatingSchedulableJob](#) implements [Serializable](#)
Schedulable Job - a data carrier class for a scheduled job that can be fired set number of times

Constructors

RepeatingSchedulableJob

public [RepeatingSchedulableJob](#) ()

RepeatingSchedulableJob

public [RepeatingSchedulableJob](#) ([MotechEvent](#) motechEvent, [Date](#) startTime, [Date](#) endTime, [Integer](#) repeatCount, [Long](#) repeatIntervalInMilliseconds, boolean ignorePastFiresAtStart)

RepeatingSchedulableJob

public [RepeatingSchedulableJob](#) ([MotechEvent](#) motechEvent, [Date](#) startTime, [Date](#) endTime, [Long](#) repeatIntervalInMilliseconds, boolean ignorePastFiresAtStart)

Methods

equals

public boolean [equals](#) ([Object](#) obj)

getEndTime

```
public Date getEndTime ()
```

getMotechEvent

```
public MotechEvent getMotechEvent ()
```

getRepeatCount

```
public Integer getRepeatCount ()
```

getRepeatIntervalInMilliseconds

```
public Long getRepeatIntervalInMilliseconds ()
```

getStartTime

```
public Date getStartTime ()
```

hashCode

```
public int hashCode ()
```

isIgnorePastFiresAtStart

```
public boolean isIgnorePastFiresAtStart ()
```

isUseOriginalFireTimeAfterMisfire

```
public boolean isUseOriginalFireTimeAfterMisfire ()
```

setEndTime

```
public RepeatingSchedulableJob setEndTime (Date endTime)
```

setIgnorePastFiresAtStart

```
public RepeatingSchedulableJob setIgnorePastFiresAtStart (boolean ignorePastFiresAtStart)
```

Ignore past fires when start time of job is in past.

```
ex : repeating job with interval of 5 unit, and current time in between fire 2 and 3 will start
  1      2      3      4
  |-----|-----|-----|
  start      ^current time
```

Parameters

- `ignorePastFiresAtStart` –

`setMotechEvent`

```
public RepeatingSchedulableJob setMotechEvent (MotechEvent motechEvent)
```

`setRepeatCount`

```
public RepeatingSchedulableJob setRepeatCount (Integer repeatCount)
```

`setRepeatIntervalInMilliseconds`

```
public RepeatingSchedulableJob setRepeatIntervalInMilliseconds (Long repeatIntervalInMil-  
liSeconds)
```

`setStartTime`

```
public RepeatingSchedulableJob setStartTime (Date startTime)
```

`setUseOriginalFireTimeAfterMisfire`

```
public RepeatingSchedulableJob setUseOriginalFireTimeAfterMisfire (boolean useOriginalFire-  
TimeAfterMisfire)
```

`toString`

```
public String toString ()
```

7.59.10 RunOnceJobId

```
public class RunOnceJobId extends JobId
```

Fields

`SUFFIX_RUNONCEJOBID`

```
public static final String SUFFIX_RUNONCEJOBID
```

Constructors

`RunOnceJobId`

```
public RunOnceJobId (String subject, String id)
```


RunOnceJobId

public **RunOnceJobId** ([MotechEvent](#) *runOnceEvent*)

7.59.11 RunOnceSchedulableJob

public final class **RunOnceSchedulableJob** implements [Serializable](#)

Run Once Schedulable Job - a data carrier class for a job scheduled in the future that can be fired only once

This class is immutable

User: Igor (iopushnyev@2paths.com) Date: 16/02/11 Time: 1:43 PM

Constructors

RunOnceSchedulableJob

public **RunOnceSchedulableJob** ([MotechEvent](#) *motechEvent*, [Date](#) *startDate*)

Constructor

Parameters

- **motechEvent** –
 - event data message that will be send by Motech Scheduler when this job is fired
- **startDate** –
 - date and time when the job fill be fired

Throws

- **IllegalArgumentException** – if *motechEvent* or *startDate* is null or *startDate* is in past

Methods

equals

public boolean **equals** ([Object](#) *o*)

getMotechEvent

public [MotechEvent](#) **getMotechEvent** ()

getStartDate

public [Date](#) **getStartDate** ()

hashCode

public int **hashCode** ()

`toString`

```
public String toString()
```

7.60 org.motechproject.scheduler.exception

7.60.1 MotechSchedulerException

public class **MotechSchedulerException** extends `RuntimeException`

User: Igor (iopushnyev@2paths.com) Date: 17/02/11 Time: 4:20 PM

Constructors

MotechSchedulerException

```
public MotechSchedulerException()
```

MotechSchedulerException

```
public MotechSchedulerException(String message)
```

MotechSchedulerException

```
public MotechSchedulerException(String message, Throwable cause)
```

MotechSchedulerException

```
public MotechSchedulerException(Throwable cause)
```

7.60.2 SchedulerInstantiationException

public class **SchedulerInstantiationException** extends `RuntimeException`

Constructors

SchedulerInstantiationException

```
public SchedulerInstantiationException(Throwable e)
```

7.60.3 SchedulerShutdownException

public class **SchedulerShutdownException** extends `RuntimeException`

The `SchedulerShutdownException` exception informs about that there were problems with shutdown scheduler.

Constructors

SchedulerShutdownException

```
public SchedulerShutdownException (Throwable e)
```

7.61 org.motechproject.scheduler.factory

7.61.1 MotechSchedulerFactoryBean

```
public class MotechSchedulerFactoryBean
```

The MotechSchedulerFactoryBean is used to created scheduler and start it.

Constructors

MotechSchedulerFactoryBean

```
public MotechSchedulerFactoryBean (ApplicationContext applicationContext, Properties scheduler-  
Properties)
```

Methods

getQuartzScheduler

```
public Scheduler getQuartzScheduler ()
```

getQuartzSchedulerFactoryBean

```
public SchedulerFactoryBean getQuartzSchedulerFactoryBean ()
```

init

```
public void init ()
```

shutdown

```
public void shutdown ()
```

7.62 org.motechproject.scheduler.service

7.62.1 MotechSchedulerActionProxyService

```
public interface MotechSchedulerActionProxyService
```

Methods

scheduleCronJob

void **scheduleCronJob** (String *subject*, Map<Object, Object> *parameters*, String *cronExpression*, DateTime *startTime*, DateTime *endTime*, Boolean *ignorePastFiresAtStart*)

scheduleDayOfWeekJob

void **scheduleDayOfWeekJob** (String *subject*, Map<Object, Object> *parameters*, DateTime *start*, DateTime *end*, List<Object> *days*, DateTime *time*, Boolean *ignorePastFiresAtStart*)

scheduleRepeatingJob

void **scheduleRepeatingJob** (String *subject*, Map<Object, Object> *parameters*, DateTime *startTime*, DateTime *endTime*, Integer *repeatCount*, Long *repeatIntervalInMilliseconds*, Boolean *ignorePastFiresAtStart*, Boolean *useOriginalFireTimeAfterMisfire*)

scheduleRunOnceJob

void **scheduleRunOnceJob** (String *subject*, Map<Object, Object> *parameters*, DateTime *startDate*)

unscheduleJobs

void **unscheduleJobs** (String *subject*)

7.62.2 MotechSchedulerService

public interface **MotechSchedulerService**

ingroup scheduler Motech Scheduler Service Interface provides methods to schedule reschedule and unschedule a job Set a global policy that determines trigger fire behaviour for misfired triggers. For details see quartz documentations for misfire policy do_nothing -> @see CronTrigger.MISFIRE_INSTRUCTION_DO_NOTHING fire_once_now -> @see CronTrigger.MISFIRE_INSTRUCTION_FIRE_ONCE_NOW ignore -> @see CronTrigger.MISFIRE_INSTRUCTION_IGNORE_MISFIRE_POLICY fire_now -> @see SimpleTrigger.MISFIRE_INSTRUCTION_FIRE_NOW ignore -> @see SimpleTrigger.MISFIRE_INSTRUCTION_IGNORE_MISFIRE_POLICY reschedule_next_with_existing_count -> @see SimpleTrigger.MISFIRE_INSTRUCTION_RESCHEDULE_NEXT_WITH_EXISTING_COUNT reschedule_next_with_remaining_count -> @see SimpleTrigger.MISFIRE_INSTRUCTION_RESCHEDULE_NEXT_WITH_REMAINING_COUNT reschedule_now_with_existing_count -> @see SimpleTrigger.MISFIRE_INSTRUCTION_RESCHEDULE_NOW_WITH_EXISTING_COUNT reschedule_now_with_remaining_count -> @see SimpleTrigger.MISFIRE_INSTRUCTION_RESCHEDULE_NOW_WITH_REMAINING_COUNT Scheduler can use couchdb for its job store. To enable set the following properties org.quartz.jobStore.class = org.motechproject.quartz.CouchDbStore org.quartz.jobStore.dbNameGenerator = org.motechproject.scheduler.service.impl.MultiTenantQuartzDatabaseName # provides database names for the jobstore, used in a multi-tenant environment to have separate databases for each tenant; leave blank to use a single database org.quartz.jobStore.properties = # a couchdb.properties file understood by ektorp to specify the database environment

Author Igor (iopushnyev@2paths.com) Date: 16/02/11

Fields

JOB_ID_KEY

`String JOB_ID_KEY`

Methods

getScheduledJobDetailedInfo

`JobDetailedInfo getScheduledJobDetailedInfo (JobBasicInfo jobBasicInfo)`

getScheduledJobTimings

`List<Date> getScheduledJobTimings (String subject, String externalJobId, Date startDate, Date endDate)`

getScheduledJobTimingsWithPrefix

`List<Date> getScheduledJobTimingsWithPrefix (String subject, String externalJobIdPrefix, Date startDate, Date endDate)`

getScheduledJobsBasicInfo

`List<JobBasicInfo> getScheduledJobsBasicInfo ()`

rescheduleJob

`void rescheduleJob (String subject, String externalId, String cronExpression)`

Reschedules a job with the given job ID to be fired according to the given Cron Expression Previous version of the configured Motech Scheduled Even that will be created when the job is fired remains us it was

Parameters

- **subject** –
- **externalId** –
- **cronExpression** –

safeScheduleJob

`void safeScheduleJob (CronSchedulableJob cronSchedulableJob)`

Same as `scheduleJob`, except that it would update existing job if one exists instead of creating a new one

Parameters

- **cronSchedulableJob** –

safeScheduleRepeatingJob

void **safeScheduleRepeatingJob** ([RepeatingSchedulableJob](#) *repeatingSchedulableJob*)

Same as `safeScheduleRepeatingJob` with `intervening = true`

Parameters

- `repeatingSchedulableJob` –

safeScheduleRunOnceJob

void **safeScheduleRunOnceJob** ([RunOnceSchedulableJob](#) *schedulableJob*)

Same as `scheduleRunOnceJob`, except that it would update existing job if one exists instead of creating a new one

Parameters

- `schedulableJob` –

safeUnscheduleAllJobs

void **safeUnscheduleAllJobs** ([String](#) *jobIdPrefix*)

safeUnscheduleJob

void **safeUnscheduleJob** ([String](#) *subject*, [String](#) *externalId*)

Same as `unscheduleJob` except that it would not throw an exception if the job doesn't exist

Parameters

- `subject` –
- `externalId` –

safeUnscheduleRepeatingJob

void **safeUnscheduleRepeatingJob** ([String](#) *subject*, [String](#) *externalId*)

Same as `unscheduleRepeatingJob` except that it would not throw an exception if the job doesn't exist

Parameters

- `subject` –
- `externalId` –

safeUnscheduleRunOnceJob

void **safeUnscheduleRunOnceJob** ([String](#) *subject*, [String](#) *externalId*)

Same as `unscheduleRunOnceJob` except that it would not throw an exception if the job doesn't exist

Parameters

- `subject` –
- `externalId` –

scheduleDayOfWeekJob

void **scheduleDayOfWeekJob** ([DayOfWeekSchedulableJob](#) *dayOfWeekSchedulableJob*)

Same as `safeScheduleDayOfWeekJob` with `intervening = true`

Parameters

- **dayOfWeekSchedulableJob** –

scheduleJob

void **scheduleJob** ([CronSchedulableJob](#) *cronSchedulableJob*)

Schedules the given schedulable job. The Job ID by which the job will be referencing in the future should be provided in an Instance of `MotechEvent` in `SchedulableJob` (see `MotechEvent.jobId`) If a job with the same job ID as the given exists, this job will be unscheduled and the given schedulable job will be scheduled

Parameters

- **cronSchedulableJob** –

scheduleRepeatingJob

void **scheduleRepeatingJob** ([RepeatingSchedulableJob](#) *repeatingSchedulableJob*)

Schedules the given schedulable job. The Job ID by which the job will be referencing in the future should be provided in an Instance of `MotechEvent` in `SchedulableJob` (see `MotechEvent.jobId`) If a job with the same job ID as the given exists, this job will be unscheduled and the given schedulable job will be scheduled

Parameters

- **repeatingSchedulableJob** –

scheduleRunOnceJob

void **scheduleRunOnceJob** ([RunOnceSchedulableJob](#) *schedulableJob*)

unscheduleAllJobs

void **unscheduleAllJobs** ([String](#) *jobIdPrefix*)

unscheduleJob

void **unscheduleJob** ([String](#) *subject*, [String](#) *externalId*)

Unscheduled a job with the given job ID

Parameters

- **subject** – : String representing domain operation eg. “pill-reminder”, “outbox-call” or `motechEvent.getSubject()`
- **externalId** – : domain specific id as String.

unscheduleJob

void **unscheduleJob** ([JobId](#) *job*)

unscheduleRepeatingJob

void **unscheduleRepeatingJob** ([String](#) *subject*, [String](#) *externalId*)

unscheduleRunOnceJob

void **unscheduleRunOnceJob** ([String](#) *subject*, [String](#) *externalId*)

Unschedules a run once job with the given job ID

Parameters

- **subject** – : [String](#) representing domain operation eg. “pill-reminder”, “outbox-call” or `motechEvent.getSubject()`
- **externalId** – : domain specific id as [String](#).

updateScheduledJob

void **updateScheduledJob** ([MotechEvent](#) *motechEvent*)

Updates MotechEvent data of the job defined by `jobId` in the given instance of that class

Parameters

- **motechEvent** –

7.63 org.motechproject.scheduler.service.impl

7.63.1 MotechScheduledJob

public class **MotechScheduledJob** implements [Job](#)

Methods

execute

public void **execute** ([JobExecutionContext](#) *jobExecutionContext*)

7.63.2 MotechScheduler

public final class **MotechScheduler**
ingroup scheduler

Main class that can bootstrap a Motech Scheduler

Author Igor (iopushnyev@2paths.com)

Methods

main

```
public static void main (String[] args)
```

7.63.3 MotechSchedulerActionProxyServiceImpl

public class **MotechSchedulerActionProxyServiceImpl** implements [MotechSchedulerActionProxyService](#)

Constructors

MotechSchedulerActionProxyServiceImpl

```
public MotechSchedulerActionProxyServiceImpl (MotechSchedulerService schedulerService)
```

Methods

scheduleCronJob

```
public void scheduleCronJob (String subject, Map<Object, Object> parameters, String cronExpression,  
                             DateTime startTime, DateTime endTime, Boolean ignorePastFiresAtStart)
```

scheduleDayOfWeekJob

```
public void scheduleDayOfWeekJob (String subject, Map<Object, Object> parameters, DateTime start,  
                                   DateTime end, List<Object> days, DateTime time, Boolean ignorePastFiresAtStart)
```

scheduleRepeatingJob

```
public void scheduleRepeatingJob (String subject, Map<Object, Object> parameters, DateTime startTime, DateTime endTime, Integer repeatCount, Long repeatIntervalInMilliseconds, Boolean ignorePastFiresAtStart, Boolean useOriginalFireTimeAfterMisfire)
```

scheduleRunOnceJob

```
public void scheduleRunOnceJob (String subject, Map<Object, Object> parameters, DateTime startDate)
```

unscheduleJobs

```
public void unscheduleJobs (String subject)
```

7.63.4 MotechSchedulerServiceImpl

public class **MotechSchedulerServiceImpl** implements [MotechSchedulerService](#)
Motech Scheduler Service implementation

See also: [MotechSchedulerService](#)

Fields

JOB_GROUP_NAME

public static final [String](#) **JOB_GROUP_NAME**

Constructors

MotechSchedulerServiceImpl

public **MotechSchedulerServiceImpl** ([MotechSchedulerFactoryBean](#) *motechSchedulerFactoryBean*,
[SettingsFacade](#) *schedulerSettings*)

Methods

assertArgumentNotNull

protected void **assertArgumentNotNull** ([String](#) *objectName*, [Object](#) *object*)

getScheduledJobDetailedInfo

public [JobDetailedInfo](#) **getScheduledJobDetailedInfo** ([JobBasicInfo](#) *jobBasicInfo*)

getScheduledJobTimings

public [List](#)<[Date](#)> **getScheduledJobTimings** ([String](#) *subject*, [String](#) *externalJobId*, [Date](#) *startDate*, [Date](#) *endDate*)

getScheduledJobTimingsWithPrefix

public [List](#)<[Date](#)> **getScheduledJobTimingsWithPrefix** ([String](#) *subject*, [String](#) *externalJobIdPrefix*,
[Date](#) *startDate*, [Date](#) *endDate*)

getScheduledJobsBasicInfo

public [List](#)<[JobBasicInfo](#)> **getScheduledJobsBasicInfo** ()

logObjectIfNotNull

protected void **logObjectIfNotNull** ([Object](#) *obj*)

rescheduleJob

public void **rescheduleJob** (*String subject*, *String externalId*, *String cronExpression*)

safeScheduleJob

public void **safeScheduleJob** (*CronSchedulableJob cronSchedulableJob*)

safeScheduleRepeatingJob

public void **safeScheduleRepeatingJob** (*RepeatingSchedulableJob repeatingSchedulableJob*)

safeScheduleRunOnceJob

public void **safeScheduleRunOnceJob** (*RunOnceSchedulableJob schedulableJob*)

safeUnscheduleAllJobs

public void **safeUnscheduleAllJobs** (*String jobIdPrefix*)

safeUnscheduleJob

public void **safeUnscheduleJob** (*String subject*, *String externalId*)

safeUnscheduleRepeatingJob

public void **safeUnscheduleRepeatingJob** (*String subject*, *String externalId*)

safeUnscheduleRunOnceJob

public void **safeUnscheduleRunOnceJob** (*String subject*, *String externalId*)

scheduleDayOfWeekJob

public void **scheduleDayOfWeekJob** (*DayOfWeekSchedulableJob dayOfWeekSchedulableJob*)

scheduleJob

public void **scheduleJob** (*CronSchedulableJob cronSchedulableJob*)

scheduleRepeatingJob

public void **scheduleRepeatingJob** (*RepeatingSchedulableJob repeatingSchedulableJob*)

scheduleRunOnceJob

```
public void scheduleRunOnceJob (RunOnceSchedulableJob schedulableJob)
```

unscheduleAllJobs

```
public void unscheduleAllJobs (String jobIdPrefix)
```

unscheduleJob

```
public void unscheduleJob (String subject, String externalId)
```

unscheduleJob

```
public void unscheduleJob (JobId job)
```

unscheduleRepeatingJob

```
public void unscheduleRepeatingJob (String subject, String externalId)
```

unscheduleRunOnceJob

```
public void unscheduleRunOnceJob (String subject, String externalId)
```

updateScheduledJob

```
public void updateScheduledJob (MotechEvent motechEvent)
```

7.64 org.motechproject.security.annotations

7.64.1 SecurityAnnotationBeanPostProcessor

```
public class SecurityAnnotationBeanPostProcessor implements BeanPostProcessor
```

A [BeanPostProcessor](#) used by Motech to load permissions from modules. Given a module context, it looks for [PreAuthorize](#) and [PostAuthorize](#) annotations. These annotations are then parsed using an [ExpressionParser](#). The permission names are deduced from [hasRole](#) and [hasAnyRole](#) in the annotation value. The names of permissions are then saved using the [MotechPermissionService](#). The bundle name used to construct the permission is retrieved from the application context.

Constructors

SecurityAnnotationBeanPostProcessor

```
public SecurityAnnotationBeanPostProcessor (MotechPermissionService permissionService)
```

Methods

postProcessAfterInitialization

public [Object](#) **postProcessAfterInitialization** ([Object](#) *bean*, [String](#) *beanName*)

postProcessBeforeInitialization

public [Object](#) **postProcessBeforeInitialization** ([Object](#) *bean*, [String](#) *beanName*)

processAnnotations

public void **processAnnotations** ([ApplicationContext](#) *applicationContext*)

7.65 org.motechproject.security.authentication

7.65.1 MotechAccessVoter

public class **MotechAccessVoter** implements [AccessDecisionVoter](#)<[Object](#)>

A custom [AccessDecisionVoter](#) for voting on whether a specific user has access to a particular URL. For example, a security rule can specify that the users motech and admin have access to a particular URL. This loads the metadata source with attributes for ACCESS_motech and ACCESS_admin. When a user invokes that URL, an affirmative based voting system will check whether or not the user is motech or admin. If not, they are denied permission, otherwise they are granted access.

Methods

supports

public boolean **supports** ([ConfigAttribute](#) *attribute*)

supports

public boolean **supports** ([Class](#)<?> *clazz*)

vote

public int **vote** ([Authentication](#) *authentication*, [Object](#) *object*, [Collection](#)<[ConfigAttribute](#)> *attributes*)

7.65.2 MotechBasicAuthenticationEntryPoint

public class **MotechBasicAuthenticationEntryPoint** extends [BasicAuthenticationEntryPoint](#)

An entry point for BASIC authentications, sets the correct server realm key.

Fields

SECURITY_REALM_KEY

public static final [String](#) **SECURITY_REALM_KEY**

Constructors

MotechBasicAuthenticationEntryPoint

public **MotechBasicAuthenticationEntryPoint** ([SettingsFacade](#) *settingsFacade*)

7.65.3 MotechLoginUrlAuthenticationEntryPoint

public class **MotechLoginUrlAuthenticationEntryPoint** extends [LoginUrlAuthenticationEntryPoint](#)

Methods

commence

public void **commence** ([HttpServletRequest](#) *request*, [HttpServletResponse](#) *response*, [AuthenticationException](#) *authException*)

7.65.4 MotechLogoutSuccessHandler

public class **MotechLogoutSuccessHandler** implements [LogoutHandler](#)

A logout handler for removing Motech user sessions from Motech's internally kept session handler. This is invoked when a user logs out.

Methods

logout

public void **logout** ([HttpServletRequest](#) *request*, [HttpServletResponse](#) *response*, [Authentication](#) *authentication*)

7.65.5 MotechPasswordEncoder

public class **MotechPasswordEncoder** extends [BCryptPasswordEncoder](#)

Methods

encodePassword

public [String](#) **encodePassword** ([String](#) *rawPassword*)

isPasswordValid

```
public boolean isPasswordValid (String encPassword, String rawPassword)
```

7.65.6 MotechRestBasicAuthenticationEntryPoint

```
public class MotechRestBasicAuthenticationEntryPoint extends BasicAuthenticationEntryPoint
```

A custom entry point that is invoked when there is an authentication exception within the filter. This ensures that when a user does not have login privileges and are unable to authenticate, a 401 unauthorized response is returned.

Fields**SECURITY_REALM_KEY**

```
public static final String SECURITY_REALM_KEY
```

Constructors**MotechRestBasicAuthenticationEntryPoint**

```
public MotechRestBasicAuthenticationEntryPoint (SettingsFacade settingsFacade)
```

Methods**commence**

```
public void commence (HttpServletRequest request, HttpServletResponse response, AuthenticationException  
                     authException)
```

7.65.7 MotechSuccessHandler

```
public class MotechSuccessHandler extends SavedRequestAwareAuthenticationSuccessHandler
```

Methods**onAuthenticationSuccess**

```
public void onAuthenticationSuccess (HttpServletRequest request, HttpServletResponse response,  
                                     Authentication authentication)
```

setSessionHandler

```
public void setSessionHandler (SessionHandler sessionHandler)
```

7.66 org.motechproject.security.builder

7.66.1 SecurityRuleBuilder

public class **SecurityRuleBuilder**

The security rule builder is responsible for building a SecurityFilterChain, which consists of a matcher pattern and a list of Spring security filters. The filters are created and configured base upon the security rule's settings.

Fields

NO_METHODS_REQUIRED_EXCEPTION_MESSAGE

public static final String **NO_METHODS_REQUIRED_EXCEPTION_MESSAGE**

NO_PATTERN_EXCEPTION_MESSAGE

public static final String **NO_PATTERN_EXCEPTION_MESSAGE**

NO_PROTOCOL_EXCEPTION_MESSAGE

public static final String **NO_PROTOCOL_EXCEPTION_MESSAGE**

NO_SUPPORTED_SCHEMES_EXCEPTION_MESSAGE

public static final String **NO_SUPPORTED_SCHEMES_EXCEPTION_MESSAGE**

Methods

buildSecurityChain

public synchronized SecurityFilterChain **buildSecurityChain** (MotechURLSecurityRule *securityRule*,
HTTPMethod *method*)

setAuthenticationManager

public void **setAuthenticationManager** (AuthenticationManager *authenticationManager*)

setBasicAuthenticationEntryPoint

public void **setBasicAuthenticationEntryPoint** (AuthenticationEntryPoint *basicAuthenticationEn-*
tryPoint)

setChannelDecisionManager

public void **setChannelDecisionManager** (ChannelDecisionManager *channelDecisionManager*)

setLoginAuthenticationEntryPoint

```
public void setLoginAuthenticationEntryPoint (AuthenticationEntryPoint loginAuthenticationEn-  
tryPoint)
```

setMotechLogoutHandler

```
public void setMotechLogoutHandler (MotechLogoutSuccessHandler motechLogoutHandler)
```

setOpenIDAuthenticationFilter

```
public void setOpenIDAuthenticationFilter (OpenIDAuthenticationFilter openIDAuthentication-  
Filter)
```

setSettingsFacade

```
public void setSettingsFacade (SettingsFacade settingsFacade)
```

setUsernamePasswordAuthenticationFilter

```
public void setUsernamePasswordAuthenticationFilter (UsernamePasswordAuthenticationFilter  
usernamePasswordAuthenticationFil-  
ter)
```

7.67 org.motechproject.security.constants

7.67.1 HTTPMethod

```
public enum HTTPMethod  
    Enumeration of HTTP request method.
```

Enum Constants**ANY**

```
public static final HTTPMethod ANY
```

DELETE

```
public static final HTTPMethod DELETE
```

GET

```
public static final HTTPMethod GET
```

HEAD

public static final [HTTPMethod](#) **HEAD**

OPTIONS

public static final [HTTPMethod](#) **OPTIONS**

POST

public static final [HTTPMethod](#) **POST**

PUT

public static final [HTTPMethod](#) **PUT**

TRACE

public static final [HTTPMethod](#) **TRACE**

7.67.2 PermissionNames

public final class **PermissionNames**

Fields

ACTIVATE_USER_PERMISSION

public static final [String](#) **ACTIVATE_USER_PERMISSION**

ADD_USER_PERMISSION

public static final [String](#) **ADD_USER_PERMISSION**

DELETE_USER_PERMISSION

public static final [String](#) **DELETE_USER_PERMISSION**

EDIT_USER_PERMISSION

public static final [String](#) **EDIT_USER_PERMISSION**

MANAGE_PERMISSION_PERMISSION

public static final [String](#) **MANAGE_PERMISSION_PERMISSION**

MANAGE_ROLE_PERMISSION

public static final [String](#) **MANAGE_ROLE_PERMISSION**

MANAGE_USER_PERMISSION

public static final [String](#) **MANAGE_USER_PERMISSION**

VIEW_BASIC_EMAIL_LOGS_PERMISSION

public static final [String](#) **VIEW_BASIC_EMAIL_LOGS_PERMISSION**

VIEW_DETAILED_EMAIL_LOGS_PERMISSION

public static final [String](#) **VIEW_DETAILED_EMAIL_LOGS_PERMISSION**

VIEW_SECURITY

public static final [String](#) **VIEW_SECURITY**

7.67.3 Protocol

public enum **Protocol**

Enumeration of Protocols supported by the module

Enum Constants

HTTP

public static final [Protocol](#) **HTTP**

HTTPS

public static final [Protocol](#) **HTTPS**

7.67.4 Scheme

public enum **Scheme**

Enumeration of scheme supported by the module

Enum Constants

BASIC

public static final [Scheme](#) **BASIC**

NO_SECURITY

public static final [Scheme](#) **NO_SECURITY**

OATH

public static final [Scheme](#) **OATH**
not supported yet

OPEN_ID

public static final [Scheme](#) **OPEN_ID**

USERNAME_PASSWORD

public static final [Scheme](#) **USERNAME_PASSWORD**

7.67.5 SecurityConfigConstants

public final class **SecurityConfigConstants**

A class for holding constants related to the options available for dynamic security rules. `MotechURLSecurityRule` is where these options are used. Prefixes related to security voting are also stored in this class.

Fields

ANY_PATTERN

public static final [String](#) **ANY_PATTERN**

ROLE_ACCESS_PREFIX

public static final [String](#) **ROLE_ACCESS_PREFIX**

USER_ACCESS_PREFIX

public static final [String](#) **USER_ACCESS_PREFIX**

7.68 org.motechproject.security.domain

7.68.1 MotechPermission

public class **MotechPermission**

Constructors

MotechPermission

```
public MotechPermission ()
```

MotechPermission

```
public MotechPermission (String permissionName, String bundleName)
```

Methods

getBundleName

```
public String getBundleName ()
```

getPermissionName

```
public String getPermissionName ()
```

setBundleName

```
public void setBundleName (String bundleName)
```

setPermissionName

```
public void setPermissionName (String permissionName)
```

toString

```
public String toString ()
```

7.68.2 MotechRole

```
public class MotechRole
```

Constructors

MotechRole

```
public MotechRole ()
```

MotechRole

```
public MotechRole (String roleName, List<String> permissionNames, boolean deletable)
```

Methods

getPermissionNames

```
public List<String> getPermissionNames ()
```

getRoleName

```
public String getRoleName ()
```

hasPermission

```
public boolean hasPermission (String permissionName)
```

isDeletable

```
public boolean isDeletable ()
```

removePermission

```
public void removePermission (String permissionName)
```

setDeletable

```
public void setDeletable (boolean deletable)
```

setPermissionNames

```
public void setPermissionNames (List<String> permissionNames)
```

setRoleName

```
public void setRoleName (String roleName)
```

7.68.3 MotechSecurityConfiguration

```
public class MotechSecurityConfiguration
```

The MotechSecurityConfiguration is a single document that contains all of the URL security rule configuration. The configuration was designed as one document because the entire filter chain must be reconstructed each time it is updated, therefore managing many references is unnecessary.

Constructors

MotechSecurityConfiguration

```
public MotechSecurityConfiguration ()
```

MotechSecurityConfiguration

```
public MotechSecurityConfiguration (List<MotechURLSecurityRule> securityRules)
```

Methods

getSecurityRules

```
public List<MotechURLSecurityRule> getSecurityRules ()
```

setSecurityRules

```
public void setSecurityRules (List<MotechURLSecurityRule> securityRules)
```

7.68.4 MotechURLSecurityRule

```
public class MotechURLSecurityRule
```

The MotechURLSecurityRule specifies the configuration for setting up a Spring SecurityFilterChain.

Details regarding configuration:

- pattern** - URL pattern the security rule applies to
- supportedSchemes** - Security rules that should apply to the URL, such as BASIC or OPEN_ID
- protocol** - Protocol the security rule applies to, such as HTTP or HTTPS
- permissionAccess** - Requires user has at least one of the listed permission to access the URL
- userAccess** - User specific access for a URL, such as motech or admin, when combined with permission access they act as an either or (one must be true)
- priority** - For future use in determining the ordering of filter chains, may be deprecated depending on UI implementation
- rest** - Whether the endpoint is meant for a form login process or as an REST end-point that does not create a session for the
- origin** - The module or user the rule originated from
- version** - The version of the module or platform the rule was created
- methodsRequired** - HTTP methods the rule applies to, if ANY is used then any method is matched, if a set is used, such as GET, POST, etc, then each will have its own corresponding filter chain with the same security found in that rule

Methods

getId

```
public Long getId ()
```

getMethodsRequired

```
public List<HTTPMethod> getMethodsRequired ()
```

getOrigin

```
public String getOrigin ()
```

getPattern

```
public String getPattern ()
```

getPermissionAccess

```
public List<String> getPermissionAccess ()
```

getPriority

```
public int getPriority ()
```

getProtocol

```
public Protocol getProtocol ()
```

getSupportedSchemes

```
public List<Scheme> getSupportedSchemes ()
```

getUserAccess

```
public List<String> getUserAccess ()
```

getVersion

```
public String getVersion ()
```

isActive

```
public boolean isActive ()
```

isDeleted

```
public boolean isDeleted ()
```


isRest

public boolean **isRest** ()

setActive

public void **setActive** (boolean *active*)

setDeleted

public void **setDeleted** (boolean *deleted*)

setId

public void **setId** (Long *id*)

setMethodsRequired

public void **setMethodsRequired** (List<HTTPMethod> *methodsRequired*)

setOrigin

public void **setOrigin** (String *origin*)

setPattern

public void **setPattern** (String *pattern*)

setPermissionAccess

public void **setPermissionAccess** (List<String> *permissionAccess*)

setPriority

public void **setPriority** (int *priority*)

setProtocol

public void **setProtocol** (Protocol *protocol*)

setRest

public void **setRest** (boolean *rest*)

setSupportedSchemes

public void **setSupportedSchemes** ([List<Scheme>](#) *supportedSchemes*)

setUserAccess

public void **setUserAccess** ([List<String>](#) *userAccess*)

setVersion

public void **setVersion** ([String](#) *version*)

7.68.5 MotechUser

public class **MotechUser**

Constructors

MotechUser

public **MotechUser** ()

MotechUser

public **MotechUser** ([String](#) *userName*, [String](#) *password*, [String](#) *email*, [String](#) *externalId*, [List<String>](#) *roles*,
[String](#) *openId*, [Locale](#) *locale*)

Methods

equals

public boolean **equals** ([Object](#) *o*)

getEmail

public [String](#) **getEmail** ()

getExternalId

public [String](#) **getExternalId** ()

getLocale

public [Locale](#) **getLocale** ()

getOpenId

```
public String getOpenId ()
```

getPassword

```
public String getPassword ()
```

getRoles

```
public List<String> getRoles ()
```

getUserName

```
public String getUserName ()
```

hasRole

```
public boolean hasRole (String role)
```

hashCode

```
public int hashCode ()
```

isActive

```
public boolean isActive ()
```

setActive

```
public void setActive (boolean active)
```

setEmail

```
public void setEmail (String email)
```

setExternalId

```
public void setExternalId (String externalId)
```

setLocale

```
public void setLocale (Locale locale)
```

setOpenId

public void **setOpenId** (*String openId*)

setPassword

public void **setPassword** (*String password*)

setRoles

public void **setRoles** (*List<String> roles*)

setUserName

public void **setUserName** (*String userName*)

toString

public *String* **toString** ()

7.68.6 MotechUserProfile

public class **MotechUserProfile** implements *Serializable*

Constructors

MotechUserProfile

public **MotechUserProfile** (*MotechUser user*)

Methods

getExternalId

public *String* **getExternalId** ()

getLocale

public *Locale* **getLocale** ()

getRoles

public *List<String>* **getRoles** ()

getUserName

```
public String getUsername ()
```

hasRole

```
public boolean hasRole (String role)
```

isActive

```
public boolean isActive ()
```

7.68.7 PasswordRecovery

```
public class PasswordRecovery
```

Methods**getEmail**

```
public String getEmail ()
```

getExpirationDate

```
public DateTime getExpirationDate ()
```

getLocale

```
public Locale getLocale ()
```

getToken

```
public String getToken ()
```

getUsername

```
public String getUsername ()
```

setEmail

```
public void setEmail (String email)
```

setExpirationDate

```
public void setExpirationDate (DateTime expirationDate)
```

setLocale

public void **setLocale** (*Locale locale*)

setToken

public void **setToken** (*String token*)

setUsername

public void **setUsername** (*String username*)

7.69 org.motechproject.security.ex

7.69.1 EmailExistsException

public class **EmailExistsException** extends *RuntimeException*

Constructors

EmailExistsException

public **EmailExistsException** (*String message*)

7.69.2 InvalidTokenException

public class **InvalidTokenException** extends *Exception*

Constructors

InvalidTokenException

public **InvalidTokenException** ()

InvalidTokenException

public **InvalidTokenException** (*String message*)

7.69.3 NonAdminUserException

public class **NonAdminUserException** extends *Exception*

Constructors

NonAdminUserException

```
public NonAdminUserException (String message)
```

7.69.4 RoleHasUserException

```
public class RoleHasUserException extends RuntimeException  
    Represents a failed attempt to delete a role currently assigned to a user.
```

Constructors

RoleHasUserException

```
public RoleHasUserException (String message)
```

7.69.5 SecurityConfigException

```
public class SecurityConfigException extends RuntimeException  
    A runtime exception thrown when the security config does not pass validation constraints required in order to  
    construct a new security chain. Ideally should not be thrown as the UI should not allow invalid data to be  
    submitted.
```

Constructors

SecurityConfigException

```
public SecurityConfigException (String message)
```

7.69.6 UserNotFoundException

```
public class UserNotFoundException extends Exception
```

Constructors

UserNotFoundException

```
public UserNotFoundException ()
```

UserNotFoundException

```
public UserNotFoundException (String message)
```

7.70 org.motechproject.security.model

7.70.1 PermissionDto

public class **PermissionDto** implements [Serializable](#)
The `PermissionDto` contains information about permission.

Constructors

PermissionDto

```
public PermissionDto ()
```

PermissionDto

```
public PermissionDto (MotechPermission motechPermission)
```

PermissionDto

```
public PermissionDto (String permissionName, String bundleName)
```

Methods

equals

```
public boolean equals (Object obj)
```

getBundleName

```
public String getBundleName ()
```

getPermissionName

```
public String getPermissionName ()
```

hashCode

```
public int hashCode ()
```

setBundleName

```
public void setBundleName (String bundleName)
```


setPermissionName

```
public void setPermissionName (String permissionName)
```

toString

```
public String toString ()
```

7.70.2 RoleDto

```
public class RoleDto
```

Transfer Motech role data between representations.

Role data transfer object facilitates exchange of role data among services, repository, and client user interface.

Constructors**RoleDto**

```
public RoleDto ()
```

RoleDto

```
public RoleDto (MotechRole motechRole)
```

RoleDto

```
public RoleDto (String roleName, List<String> permissionNames)
```

RoleDto

```
public RoleDto (String roleName, List<String> permissionNames, boolean deletable)
```

Methods**equals**

```
public boolean equals (Object obj)
```

getOriginalRoleName

```
public String getOriginalRoleName ()
```

getPermissionNames

```
public List<String> getPermissionNames ()
```

getRoleName

```
public String getRoleName ()
```

hashCode

```
public int hashCode ()
```

isDeletable

```
public boolean isDeletable ()
```

setDeletable

```
public void setDeletable (boolean deletable)
```

setOriginalRoleName

```
public void setOriginalRoleName (String originalRoleName)
```

setPermissionNames

```
public void setPermissionNames (List<String> permissionNames)
```

setRoleName

```
public void setRoleName (String roleName)
```

toString

```
public String toString ()
```

7.70.3 SecurityConfigDto

```
public class SecurityConfigDto
```

Used to transfer security configuration to and from a web request and UI

Methods

getSecurityRules

```
public List<SecurityRuleDto> getSecurityRules ()
```

setSecurityRules

```
public void setSecurityRules (List<SecurityRuleDto> securityRules)
```

7.70.4 SecurityRuleDto

```
public class SecurityRuleDto
```

Transfer Motech security rule data between representations.

Methods**getId**

```
public Long getId ()
```

getMethodsRequired

```
public List<String> getMethodsRequired ()
```

getOrigin

```
public String getOrigin ()
```

getPattern

```
public String getPattern ()
```

getPermissionAccess

```
public List<String> getPermissionAccess ()
```

getPriority

```
public int getPriority ()
```

getProtocol

```
public String getProtocol ()
```

getSupportedSchemes

```
public List<String> getSupportedSchemes ()
```

getUserAccess

public [List<String>](#) **getUserAccess** ()

getVersion

public [String](#) **getVersion** ()

isActive

public boolean **isActive** ()

isDeleted

public boolean **isDeleted** ()

isRest

public boolean **isRest** ()

setActive

public void **setActive** (boolean *active*)

setDeleted

public void **setDeleted** (boolean *deleted*)

setId

public void **setId** ([Long](#) *id*)

setMethodsRequired

public void **setMethodsRequired** ([List<String>](#) *methodsRequired*)

setOrigin

public void **setOrigin** ([String](#) *origin*)

setPattern

public void **setPattern** ([String](#) *pattern*)

setPermissionAccess

```
public void setPermissionAccess (List<String> permissionAccess)
```

setPriority

```
public void setPriority (int priority)
```

setProtocol

```
public void setProtocol (String protocol)
```

setRest

```
public void setRest (boolean rest)
```

setSupportedSchemes

```
public void setSupportedSchemes (List<String> supportedSchemes)
```

setUserAccess

```
public void setUserAccess (List<String> userAccess)
```

setVersion

```
public void setVersion (String version)
```

7.70.5 UserDto

```
public class UserDto
```

Constructors**UserDto**

```
public UserDto ()
```

UserDto

```
public UserDto (MotechUser motechUser)
```

Methods

`getEmail`

```
public String getEmail ()
```

`getExternalId`

```
public String getExternalId ()
```

`getLocale`

```
public Locale getLocale ()
```

`getOpenId`

```
public String getOpenId ()
```

`getPassword`

```
public String getPassword ()
```

`getRoles`

```
public List<String> getRoles ()
```

`getUserName`

```
public String getUserName ()
```

`isActive`

```
public boolean isActive ()
```

`isGeneratePassword`

```
public boolean isGeneratePassword ()
```

`setActive`

```
public void setActive (boolean active)
```

`setEmail`

```
public void setEmail (String email)
```

setExternalId

public void **setExternalId** (*String externalId*)

setGeneratePassword

public void **setGeneratePassword** (boolean *generatePassword*)

setLocale

public void **setLocale** (*Locale locale*)

setOpenId

public void **setOpenId** (*String openId*)

setPassword

public void **setPassword** (*String password*)

setRoles

public void **setRoles** (*List<String> roles*)

setUserName

public void **setUserName** (*String userName*)

7.71 org.motechproject.security.repository

7.71.1 AllMotechPermissions

public class **AllMotechPermissions**

Methods**add**

public void **add** (*MotechPermission permission*)

delete

public void **delete** (*MotechPermission permission*)

findByPermissionName

```
public MotechPermission findByPermissionName (String permissionName)
```

getPermissions

```
public List<MotechPermission> getPermissions ()
```

setDataService

```
public void setDataService (MotechPermissionsDataService dataService)
```

7.71.2 AllMotechRoles

```
public class AllMotechRoles
```

Methods

add

```
public void add (MotechRole role)
```

findByRoleName

```
public MotechRole findByRoleName (String roleName)
```

getRoles

```
public List<MotechRole> getRoles ()
```

remove

```
public void remove (MotechRole motechRole)
```

setDataService

```
public void setDataService (MotechRolesDataService dataService)
```

update

```
public void update (MotechRole motechRole)
```


7.71.3 AllMotechSecurityRules

public class **AllMotechSecurityRules**

Implementation of DAO interface that utilizes a MDS back-end for storage. Only one MotechSecurityConfiguration file should be saved at a time, so adding the document looks for the old document in order to update it if it already exists. Rather than updating the object reference, the old configuration's ID and revision are used for the new document.

Methods

addOrUpdate

public void **addOrUpdate** (*MotechSecurityConfiguration config*)

getMotechSecurityConfiguration

public *MotechSecurityConfiguration* **getMotechSecurityConfiguration** ()

getRuleById

public *MotechURLSecurityRule* **getRuleById** (*Long id*)

getRules

public *List*<*MotechURLSecurityRule*> **getRules** ()

getRulesByOrigin

public *List*<*MotechURLSecurityRule*> **getRulesByOrigin** (*String origin*)

remove

public void **remove** (*MotechSecurityConfiguration config*)

setDataService

public void **setDataService** (*MotechURLSecurityRuleDataService dataService*)

7.71.4 AllMotechUsers

public class **AllMotechUsers**

Methods

add

public void **add** (*MotechUser user*)

addOpenIdUser

public void **addOpenIdUser** (*MotechUser user*)

findByRole

public List<*MotechUser*> **findByRole** (*String role*)

findByUsername

public *MotechUser* **findByUsername** (*String userName*)

findUserByEmail

public *MotechUser* **findUserByEmail** (*String email*)

findUserByOpenId

public *MotechUser* **findUserByOpenId** (*String openId*)

getOpenIdUsers

public List<*MotechUser*> **getOpenIdUsers** ()

getUsers

public List<*MotechUser*> **getUsers** ()

remove

public void **remove** (*MotechUser motechUser*)

setDataService

public void **setDataService** (*MotechUsersDataService dataService*)

update

public void **update** (*MotechUser motechUser*)

7.71.5 AllPasswordRecoveries

public class **AllPasswordRecoveries**

Methods

add

public void **add** ([PasswordRecovery](#) *passwordRecovery*)

allRecoveries

public [List](#)<[PasswordRecovery](#)> **allRecoveries** ()

createRecovery

public [PasswordRecovery](#) **createRecovery** ([String](#) *username*, [String](#) *email*, [String](#) *token*, [DateTime](#) *expirationDate*, [Locale](#) *locale*)

findForToken

public [PasswordRecovery](#) **findForToken** ([String](#) *token*)

findForUser

public [PasswordRecovery](#) **findForUser** ([String](#) *username*)

getExpired

public [List](#)<[PasswordRecovery](#)> **getExpired** ()

remove

public void **remove** ([PasswordRecovery](#) *passwordRecovery*)

setDataService

public void **setDataService** ([PasswordRecoveriesDataService](#) *dataService*)

update

public void **update** ([PasswordRecovery](#) *passwordRecovery*)

7.71.6 MotechPermissionsDataService

public interface **MotechPermissionsDataService** extends [MotechDataService](#)<[MotechPermission](#)>

Methods

findByPermissionName

[MotechPermission](#) **findByPermissionName** ([String](#) *permissionName*)

7.71.7 MotechRolesDataService

public interface **MotechRolesDataService** extends [MotechDataService](#)<[MotechRole](#)>

Methods

findByRoleName

[MotechRole](#) **findByRoleName** ([String](#) *roleName*)

7.71.8 MotechURLSecurityRuleDataService

public interface **MotechURLSecurityRuleDataService** extends [MotechDataService](#)<[MotechURLSecurityRule](#)>

Methods

findByOrigin

[List](#)<[MotechURLSecurityRule](#)> **findByOrigin** ([String](#) *origin*)

7.71.9 MotechUsersDataService

public interface **MotechUsersDataService** extends [MotechDataService](#)<[MotechUser](#)>

Methods

findByEmail

[MotechUser](#) **findByEmail** ([String](#) *email*)

findByOpenId

[MotechUser](#) **findByOpenId** ([String](#) *openId*)

findByRole

List<MotechUser> **findByRole** (String *role*)

findByUserName

MotechUser **findByUserName** (String *userName*)

7.71.10 PasswordRecoveriesDataService

public interface **PasswordRecoveriesDataService** extends MotechDataService<PasswordRecovery>

Methods**findByExpirationDate**

List<PasswordRecovery> **findByExpirationDate** (Range<DateTime> *range*)

findForToken

PasswordRecovery **findForToken** (String *token*)

findForUser

PasswordRecovery **findForUser** (String *username*)

7.72 org.motechproject.security.service

7.72.1 AuthoritiesService

public interface **AuthoritiesService**
Service interface to retrieve authorities(permissions) for a given MotechUser

Methods**authoritiesFor**

List<GrantedAuthority> **authoritiesFor** (MotechUser *user*)

7.72.2 MotechPermissionService

public interface **MotechPermissionService**
Service for managing Motech permissions.

Methods

addPermission

void **addPermission** (*PermissionDto permission*)

deletePermission

void **deletePermission** (*String permissionName*)

getPermissions

List<PermissionDto> **getPermissions** ()

7.72.3 MotechProxyManager

public class **MotechProxyManager**

The MotechProxyManager acts as a wrapper around Spring's FilterChainProxy. The FilterChainProxy contains a list of immutable SecurityFilterChain objects which Spring's security consults for filters when handling requests. In order to dynamically define new secure, a new FilterChainProxy is constructed and the reference is updated. The MotechProxyManager acts as a customized delegate in MotechDelegatingFilterProxy.

Methods

getDefaultSecurityConfiguration

public *MotechSecurityConfiguration* **getDefaultSecurityConfiguration** ()

This method reads default security configuration from the file containing security rules and returns it.

Returns MotechSecurityConfiguration default security rules

getFilterChainProxy

public *FilterChainProxy* **getFilterChainProxy** ()

initializeProxyChain

public void **initializeProxyChain** ()

This method serves the same purpose of rebuildProxyChain, but does not require any kind of security authentication so it should only ever be used by the activator, which does not have an authentication object.

rebuildProxyChain

public synchronized void **rebuildProxyChain** ()

Method to invoke to dynamically re-define the Spring security. All rules converted into security filter chains in order to create a new FilterChainProxy. The order of the rules in the list matters for filtering purposes.

setProxy

public void **setProxy** (*FilterChainProxy proxy*)

setSecurityRuleBuilder

public void **setSecurityRuleBuilder** (*SecurityRuleBuilder securityRuleBuilder*)

setSecurityRulesDAO

public void **setSecurityRulesDAO** (*AllMotechSecurityRules securityRulesDAO*)

7.72.4 MotechRoleService

public interface **MotechRoleService**

Methods**createRole**

void **createRole** (*RoleDto role*)

deleteRole

void **deleteRole** (*RoleDto role*)

getRole

RoleDto **getRole** (*String roleName*)

getRoles

List<RoleDto> **getRoles** ()

updateRole

void **updateRole** (*RoleDto role*)

7.72.5 MotechURLSecurityService

public interface **MotechURLSecurityService**

Service to access and update security configuration details from the platform. Permission based, method level security is defined to prevent unauthorized users from updating security.

Methods

findAllSecurityRules

List<SecurityRuleDto> **findAllSecurityRules** ()

A protected method for viewing security rule information for the platform.

Returns All URL security rules found in the database

updateSecurityConfiguration

void **updateSecurityConfiguration** (SecurityConfigDto *configuration*)

A protected method for updating security configuration for the platform.

Parameters

- **configuration** – The updated security information, which will cause an updating of the motech proxy manager

7.72.6 MotechUserService

public interface **MotechUserService**

Service interface that defines APIs to retrieve and manage user details

Methods

activateUser

void **activateUser** (String *username*)

changePassword

MotechUserProfile **changePassword** (String *username*, String *oldPassword*, String *newPassword*)

deleteUser

void **deleteUser** (UserDto *user*)

getCurrentUser

UserDto **getCurrentUser** ()

getLocale

Locale **getLocale** (String *userName*)

getOpenIdUsers

List<MotechUserProfile> **getOpenIdUsers** ()

getRoles

List<String> **getRoles** (String *userName*)

getUser

UserDto **getUser** (String *userName*)

getUserByEmail

UserDto **getUserByEmail** (String *email*)

getUsers

List<MotechUserProfile> **getUsers** ()

hasActiveAdminUser

boolean **hasActiveAdminUser** ()

hasUser

boolean **hasUser** (String *username*)

register

void **register** (String *username*, String *password*, String *email*, String *externalId*, List<String> *roles*, Locale *locale*)

register

void **register** (String *username*, String *password*, String *email*, String *externalId*, List<String> *roles*, Locale *locale*, boolean *isActive*, String *openId*)

retrieveUserByCredentials

MotechUserProfile **retrieveUserByCredentials** (String *username*, String *password*)

sendLoginInformation

void **sendLoginInformation** (String *userName*, String *password*)

setLocale

void **setLocale** (*String* *userName*, *Locale* *locale*)

updateUserDetailsWithPassword

void **updateUserDetailsWithPassword** (*UserDto* *user*)

updateUserDetailsWithoutPassword

void **updateUserDetailsWithoutPassword** (*UserDto* *user*)

7.72.7 PasswordRecoveryService

public interface **PasswordRecoveryService**

Methods

cleanUpExpiredRecoveries

void **cleanUpExpiredRecoveries** ()

oneTimeTokenOpenId

void **oneTimeTokenOpenId** (*String* *email*)

passwordRecoveryRequest

void **passwordRecoveryRequest** (*String* *email*)

resetPassword

void **resetPassword** (*String* *token*, *String* *password*, *String* *passwordConfirmation*)

validateToken

boolean **validateToken** (*String* *token*)

validateTokenAndLoginUser

void **validateTokenAndLoginUser** (*String* *token*, *HttpServletRequest* *request*, *HttpServletResponse* *response*)

7.72.8 SecurityRoleLoader

public class **SecurityRoleLoader**

Constructors

SecurityRoleLoader

public **SecurityRoleLoader** ([MotechRoleService](#) *roleService*)

Methods

loadRoles

public void **loadRoles** ([ApplicationContext](#) *applicationContext*)

7.72.9 SecurityRuleLoader

public class **SecurityRuleLoader**

Helper class that scans an application context for security rules and re-initializes the MotechProxyManager security chain.

Methods

loadRules

public synchronized void **loadRules** ([ApplicationContext](#) *applicationContext*)

Attempts to load rules from the application context, if rules are found, the security configuration is updated. Synchronized so there are not race conditions on the data.

setAllSecurityRules

public void **setAllSecurityRules** ([AllMotechSecurityRules](#) *allSecurityRules*)

setProxyManager

public void **setProxyManager** ([MotechProxyManager](#) *proxyManager*)

7.72.10 UserContextService

public interface **UserContextService**

Interface to refresh user context (all or specified username)

Methods

refreshAllUsersContextIfActive

```
void refreshAllUsersContextIfActive ()
```

refreshUserContextIfActive

```
void refreshUserContextIfActive (String userName)
```

7.73 org.motechproject.server.api

7.73.1 BundleIcon

```
public class BundleIcon
```

Fields

ICON_LOCATIONS

```
public static final String[] ICON_LOCATIONS
```

Constructors

BundleIcon

```
public BundleIcon (byte[] icon, String mime)
```

Methods

getContentLength

```
public int getContentLength ()
```

getIcon

```
public byte[] getIcon ()
```

getMime

```
public String getMime ()
```

7.73.2 BundleInformation

public class **BundleInformation**

Class acting as a DTO for a [Bundle](#) in the system.

Fields

BUNDLE_NAME

protected static final [String](#) **BUNDLE_NAME**

Constructors

BundleInformation

public **BundleInformation** ([Bundle](#) *bundle*)

Methods

equals

public boolean **equals** ([Object](#) *arg0*)

getAngularModule

public [String](#) **getAngularModule** ()

getBundleId

public long **getBundleId** ()

getLocation

public [String](#) **getLocation** ()

getModuleName

public [String](#) **getModuleName** ()

getName

public [String](#) **getName** ()

getSettingsURL

```
public String getSettingsURL ()
```

getState

```
public State getState ()
```

getSymbolicName

```
public String getSymbolicName ()
```

getVersion

```
public Version getVersion ()
```

hasStatus

```
public boolean hasStatus (int status)
```

hashCode

```
public int hashCode ()
```

setAngularModule

```
public void setAngularModule (String angularModule)
```

setModuleName

```
public void setModuleName (String moduleName)
```

setSettingsURL

```
public void setSettingsURL (String settingsURL)
```

7.73.3 BundleInformation.State

```
public enum State
```

Represents the bundle state.

Enum Constants

ACTIVE

public static final [BundleInformation.State](#) **ACTIVE**

INSTALLED

public static final [BundleInformation.State](#) **INSTALLED**

RESOLVED

public static final [BundleInformation.State](#) **RESOLVED**

STARTING

public static final [BundleInformation.State](#) **STARTING**

STOPPING

public static final [BundleInformation.State](#) **STOPPING**

UNINSTALLED

public static final [BundleInformation.State](#) **UNINSTALLED**

UNKNOWN

public static final [BundleInformation.State](#) **UNKNOWN**

7.73.4 BundleLoader

public interface **BundleLoader**

Interface for custom bundle loading processes

Author Ricky Wang

Methods

loadBundle

void **loadBundle** ([Bundle](#) *bundle*)

Parameters

- **bundle** –

Throws

- [BundleLoadingException](#) –

7.73.5 BundleLoadingException

public class **BundleLoadingException** extends [Exception](#)

Constructors

BundleLoadingException

public **BundleLoadingException** ([String message](#))

BundleLoadingException

public **BundleLoadingException** ([String message](#), [Throwable cause](#))

BundleLoadingException

public **BundleLoadingException** ([Throwable cause](#))

7.73.6 JarInformation

public class **JarInformation**

Fields

BUNDLE_SYMBOLIC_NAME

public static final [String](#) **BUNDLE_SYMBOLIC_NAME**

BUNDLE_VERSION

public static final [String](#) **BUNDLE_VERSION**

EXTRACTION_FAILED

public static final [String](#) **EXTRACTION_FAILED**

IMPLEMENTATION_TITLE

public static final [String](#) **IMPLEMENTATION_TITLE**

IMPLEMENTATION_VENDOR_ID

public static final [String](#) **IMPLEMENTATION_VENDOR_ID**

IMPLEMENTATION_VERSION

```
public static final String IMPLEMENTATION_VERSION
```

Constructors

JarInformation

```
public JarInformation (File file)
```

Methods

getBundleSymbolicName

```
public String getBundleSymbolicName ()
```

getBundleVersion

```
public String getBundleVersion ()
```

getFilename

```
public String getFilename ()
```

getImplementationTitle

```
public String getImplementationTitle ()
```

getImplementationVendorID

```
public String getImplementationVendorID ()
```

getImplementationVersion

```
public String getImplementationVersion ()
```

getPath

```
public String getPath ()
```

7.73.7 JarInformationHandler

```
public class JarInformationHandler
```

Fields

JAR_FILE_EXTENSION

public static final `String` **JAR_FILE_EXTENSION**

Constructors

JarInformationHandler

public **JarInformationHandler** (`String` *path*)

Methods

extractJarInformationFromPath

public void **extractJarInformationFromPath** ()

getJarList

public `List`<`JarInformation`> **getJarList** ()

getPath

public `String` **getPath** ()

initHandler

public void **initHandler** ()

7.74 org.motechproject.server.config

7.74.1 SettingsFacade

public class **SettingsFacade**

SettingsFacade provides an interface to access application configuration present in files or database

Methods

afterPropertiesSet

public void **afterPropertiesSet** ()

areConfigurationSettingsRegistered

```
public boolean areConfigurationSettingsRegistered()
```

asProperties

```
public Properties asProperties()
```

constructSymbolicName

```
protected String constructSymbolicName()
```

findFilename

```
protected String findFilename(String key)
```

getActivemqConfig

```
public Properties getActivemqConfig()
```

getModuleName

```
public String getModuleName()
```

getPlatformSettings

```
public MotechSettings getPlatformSettings()
```

getProperties

```
public Properties getProperties(String filename)
```

getProperty

```
public String getProperty(String key)
```

getProperty

```
public String getProperty(String key, String filename)
```

getRawConfig

```
public InputStream getRawConfig(String filename)
```

getResourceFileName

protected static `String` **getResourceFileName** (`Resource` *resource*)

getSymbolicName

public `String` **getSymbolicName** ()

registerAllProperties

protected void **registerAllProperties** ()

registerAllRawConfig

protected void **registerAllRawConfig** ()

registerProperties

protected void **registerProperties** (`String` *filename*, `Properties` *properties*)

saveConfigProperties

public void **saveConfigProperties** (`String` *filename*, `Properties` *properties*)

savePlatformSettings

public void **savePlatformSettings** (`MotechSettings` *settings*)

saveRawConfig

public void **saveRawConfig** (`String` *filename*, `Resource` *resource*)

setConfigFiles

public void **setConfigFiles** (`List<Resource>` *resources*)

setConfigurationService

public void **setConfigurationService** (`ConfigurationService` *configurationService*)

setModuleName

public void **setModuleName** (`String` *moduleName*)

setProperty

```
public void setProperty (String key, String value)
```

setRawConfigFiles

```
public void setRawConfigFiles (List<Resource> resources)
```

unregisterProperties

```
public void unregisterProperties (String symbolicName)
```

7.75 org.motechproject.server.config.domain

7.75.1 LoginMode

```
public final class LoginMode  
    Encapsulates the operations on login mode
```

Fields**OPEN_ID**

```
public static final LoginMode OPEN_ID
```

REPOSITORY

```
public static final LoginMode REPOSITORY
```

Methods**getName**

```
public String getName ()
```

isOpenId

```
public boolean isOpenId ()
```

isRepository

```
public boolean isRepository ()
```

valueOf

public static [LoginMode](#) **valueOf** ([String](#) *loginMode*)

7.75.2 MotechSettings

public interface **MotechSettings**

Interface for main motech settings managment

Methods

asProperties

[Properties](#) **asProperties** ()

getActivemqProperties

[Properties](#) **getActivemqProperties** ()

getConfigFileChecksum

[String](#) **getConfigFileChecksum** ()

getFilePath

[String](#) **getFilePath** ()

getLanguage

[String](#) **getLanguage** ()

getLastRun

[DateTime](#) **getLastRun** ()

getLoginMode

[LoginMode](#) **getLoginMode** ()

getProviderName

[String](#) **getProviderName** ()

getProviderUrl

`String getProviderUrl ()`

getServerHost

`String getServerHost ()`

getServerUrl

`String getServerUrl ()`

getStatusMsgTimeout

`String getStatusMsgTimeout ()`

getUploadSize

`String getUploadSize ()`

isPlatformInitialized

`boolean isPlatformInitialized ()`

load

`void load (DigestInputStream dis)`

savePlatformSetting

`void savePlatformSetting (String key, String value)`

setConfigFileChecksum

`void setConfigFileChecksum (String configFileChecksum)`

setFilePath

`void setFilePath (String filePath)`

setLanguage

`void setLanguage (String language)`

setLastRun

void **setLastRun** (*DateTime lastRun*)

setLoginModeValue

void **setLoginModeValue** (*String loginMode*)

setPlatformInitialized

void **setPlatformInitialized** (*boolean platformInitialized*)

setProviderName

void **setProviderName** (*String providerName*)

setProviderUrl

void **setProviderUrl** (*String providerUrl*)

setServerUrl

void **setServerUrl** (*String serverUrl*)

setStatusMsgTimeout

void **setStatusMsgTimeout** (*String statusMsgTimeout*)

setUploadSize

void **setUploadSize** (*String uploadSize*)

updateFromProperties

void **updateFromProperties** (*Properties props*)

updateSettings

void **updateSettings** (*String configFileChecksum, String filePath, Properties platformSettings*)

7.75.3 MotechURL

public class **MotechURL**

Fields

URL_PATTERN

public static final [String](#) **URL_PATTERN**

Constructors

MotechURL

public **MotechURL** ([String](#) *url*)

Methods

getHost

public [String](#) **getHost** ()

toString

public [String](#) **toString** ()

7.75.4 SettingsRecord

public class **SettingsRecord** implements [MotechSettings](#)
Class for storing settings values

Constructors

SettingsRecord

public **SettingsRecord** ()

Methods

asProperties

public [Properties](#) **asProperties** ()

getActivemqProperties

public [Properties](#) **getActivemqProperties** ()

getConfigFileChecksum

public [String](#) **getConfigFileChecksum** ()

getFilePath

```
public String getFilePath ()
```

getLanguage

```
public String getLanguage ()
```

getLastRun

```
public DateTime getLastRun ()
```

getLoginMode

```
public LoginMode getLoginMode ()
```

getLoginModeValue

```
public String getLoginModeValue ()
```

getPlatformSettings

```
public Map<String, String> getPlatformSettings ()
```

getProviderName

```
public String getProviderName ()
```

getProviderUrl

```
public String getProviderUrl ()
```

getServerHost

```
public String getServerHost ()
```

getServerUrl

```
public String getServerUrl ()
```

getStatusMsgTimeout

```
public String getStatusMsgTimeout ()
```

getUploadSize

```
public String getUploadSize ()
```

isPlatformInitialized

```
public boolean isPlatformInitialized ()
```

load

```
public synchronized void load (DigestInputStream dis)
```

mergeWithDefaults

```
public void mergeWithDefaults (Properties defaultConfig)
```

removeDefaults

```
public void removeDefaults (Properties defaultConfig)
```

savePlatformSetting

```
public void savePlatformSetting (String key, String value)
```

setConfigFileChecksum

```
public void setConfigFileChecksum (String configFileChecksum)
```

setFilePath

```
public void setFilePath (String filePath)
```

setLanguage

```
public void setLanguage (String language)
```

setLastRun

```
public void setLastRun (DateTime lastRun)
```

setLoginModeValue

```
public void setLoginModeValue (String loginMode)
```

setPlatformInitialized

public void **setPlatformInitialized** (boolean *platformInitialized*)

setPlatformSettings

public void **setPlatformSettings** (Map<String, String> *platformSettings*)

setProviderName

public void **setProviderName** (String *providerName*)

setProviderUrl

public void **setProviderUrl** (String *providerUrl*)

setServerUrl

public void **setServerUrl** (String *serverUrl*)

setStatusMsgTimeout

public void **setStatusMsgTimeout** (String *statusMsgTimeout*)

setUploadSize

public void **setUploadSize** (String *uploadSize*)

updateFromProperties

public void **updateFromProperties** (Properties *props*)

updateSettings

public void **updateSettings** (String *configFileChecksum*, String *filePath*, Properties *platformSettings*)

7.76 org.motechproject.server.config.service

7.76.1 ConfigLoader

public class **ConfigLoader**

Config loader used to load the platform core settings.

Methods

findExistingConfigs

public [List<File>](#) **findExistingConfigs** ()

Finds all configs from the config location

Throws

- **IOException** – If there is any error while handling the files.

loadDefaultConfig

public [SettingsRecord](#) **loadDefaultConfig** ()

loadMotechSettings

public [SettingsRecord](#) **loadMotechSettings** ()

setCoreConfigurationService

public void **setCoreConfigurationService** ([CoreConfigurationService](#) *coreConfigurationService*)

setEventAdmin

public void **setEventAdmin** ([EventAdmin](#) *eventAdmin*)

setResourceLoader

public void **setResourceLoader** ([ResourceLoader](#) *resourceLoader*)

7.76.2 PlatformSettingsService

public interface **PlatformSettingsService**

Platform Settings service used to handle platform settings.

Methods

exportPlatformSettings

[Properties](#) **exportPlatformSettings** ()

7.76.3 SettingService

public interface **SettingService** extends [MotechDataService<SettingsRecord>](#)

7.77 org.motechproject.server.startup

7.77.1 MotechPlatformState

public enum **MotechPlatformState**

Defines the different states of the MOTECH system.

Enum Constants

DB_ERROR

public static final [MotechPlatformState](#) **DB_ERROR**

FIRST_RUN

public static final [MotechPlatformState](#) **FIRST_RUN**

NEED_BOOTSTRAP_CONFIG

public static final [MotechPlatformState](#) **NEED_BOOTSTRAP_CONFIG**

NEED_CONFIG

public static final [MotechPlatformState](#) **NEED_CONFIG**

NORMAL_RUN

public static final [MotechPlatformState](#) **NORMAL_RUN**

NO_DB

public static final [MotechPlatformState](#) **NO_DB**

STARTUP

public static final [MotechPlatformState](#) **STARTUP**

7.77.2 StartupManager

public class **StartupManager**

StartupManager controlling and managing the application loading

Methods

canLaunchBundles

public boolean **canLaunchBundles** ()

getDefaultSettings

public [SettingsRecord](#) **getDefaultSettings** ()

This function is only called when the default configuration is loaded and is no config in the database or external files

isBootstrapConfigRequired

public boolean **isBootstrapConfigRequired** ()

isConfigRequired

public boolean **isConfigRequired** ()

startup

public void **startup** ()

7.78 org.motechproject.server.ui

7.78.1 LocaleService

public interface **LocaleService**

A service responsible for localization. Allows retrieval/settings of user language as well as retrieving localized messages for a user request. Can also be used to retrieve a list of usable languages.

Methods

getMessages

[Map](#)<[String](#), [String](#)> **getMessages** ([HttpServletRequest](#) *request*)

getSupportedLanguages

[NavigableMap](#)<[String](#), [String](#)> **getSupportedLanguages** ()

getUserLocale

[Locale](#) **getUserLocale** ([HttpServletRequest](#) *request*)

setUserLocale

void **setUserLocale** ([HttpServletRequest](#) *request*, [HttpServletResponse](#) *response*, [Locale](#) *locale*)

7.79 org.motechproject.server.ui.ex

7.79.1 AlreadyRegisteredException

public class **AlreadyRegisteredException** extends [RuntimeException](#)

Constructors

AlreadyRegisteredException

public **AlreadyRegisteredException** ([String](#) *msg*)

7.80 org.motechproject.server.web.controller

7.80.1 Constants

public final class **Constants**
Class that has all the common UI constants.

Fields

REDIRECT_BOOTSTRAP

public static final [String](#) **REDIRECT_BOOTSTRAP**

REDIRECT_HOME

public static final [String](#) **REDIRECT_HOME**

REDIRECT_STARTUP

public static final [String](#) **REDIRECT_STARTUP**

7.80.2 DashboardController

public class **DashboardController**
Main application controller. Responsible for retrieving information shared across the UI of different modules.
The view returned by this controller will embed the UI of the currently requested module.

Methods

accessdenied

public ModelAndView **accessdenied** (HttpServletRequest *request*)

getTime

public String **getTime** (HttpServletRequest *request*)

getUptime

public DateTime **getUptime** ()

getUser

public UserInfo **getUser** (HttpServletRequest *request*)

index

public ModelAndView **index** (HttpServletRequest *request*)

setBundleContext

public void **setBundleContext** (BundleContext *bundleContext*)

setLocaleService

public void **setLocaleService** (LocaleService *localeService*)

setStartupManager

public void **setStartupManager** (StartupManager *startupManager*)

7.80.3 ForgotController

public class **ForgotController**
Forgot Controller for reset password.

Methods

forgotPost

public String **forgotPost** (String *email*)

getForgotViewData

```
public ForgotViewData getForgotViewData (HttpServletRequest request)
```

login

```
public ModelAndView login (HttpServletRequest request)
```

7.80.4 LocaleController

```
public class LocaleController
```

The `LocaleController` class is responsible for handling requests connected with internationalization

Methods

getAvailableLocales

```
public Map<String, String> getAvailableLocales (HttpServletRequest request)
```

getLangLocalisation

```
public Map<String, String> getLangLocalisation (HttpServletRequest request)
```

getSupportedLanguages

```
public NavigableMap<String, String> getSupportedLanguages ()
```

getUserLang

```
public String getUserLang (HttpServletRequest request)
```

setUserLang

```
public void setUserLang (HttpServletRequest request, HttpServletResponse response, String language,  
                        String country, String variant)
```

7.80.5 LoginController

```
public class LoginController
```

Login Controller for user authentication.

Methods

getLoginViewData

```
public LoginViewData getLoginViewData (HttpServletRequest request)
```

login

```
public ModelAndView login (HttpServletRequest request)
```

7.80.6 ModuleController

```
public class ModuleController
```

Methods

getConfig

```
public List<ModuleConfig> getConfig ()
```

getCriticalMessage

```
public String getCriticalMessage (String moduleName)
```

getMenu

```
public ModuleMenu getMenu (HttpServletRequest request)
```

getUser

```
public UserInfo getUser (HttpServletRequest request)
```

setBundleContext

```
public void setBundleContext (BundleContext bundleContext)
```

setLocaleService

```
public void setLocaleService (LocaleService localeService)
```

setMenuBuilder

```
public void setMenuBuilder (MenuBuilder menuBuilder)
```

setUiFrameworkService

```
public void setUiFrameworkService (UiFrameworkService uiFrameworkService)
```

7.80.7 ResetController

```
public class ResetController
```

Methods

getResetViewData

```
public ResetViewData getResetViewData (HttpServletRequest request)
```

reset

```
public ResetViewData reset (ResetForm form, HttpServletRequest request)
```

resetView

```
public ModelAndView resetView (HttpServletRequest request)
```

7.80.8 StartupController

```
public class StartupController
```

```
    StartupController that manages the platform system start up and captures the platform core settings and user information.
```

Methods

getStartupViewData

```
public StartupViewData getStartupViewData (HttpServletRequest request)
```

setStartupFormValidatorFactory

```
public void setStartupFormValidatorFactory (StartupFormValidatorFactory validatorFactory)
```

startup

```
public ModelAndView startup ()
```

submitForm

```
public List<String> submitForm (StartupForm startupSettings)
```

7.80.9 StatusController

```
public class StatusController
```

Methods

status

```
public String status ()
```

7.81 org.motechproject.server.web.form

7.81.1 LoginForm

```
public class LoginForm
```

Methods

getPassword

```
public String getPassword ()
```

getUserName

```
public String getUserName ()
```

setPassword

```
public void setPassword (String password)
```

setUserName

```
public void setUsername (String userName)
```

7.81.2 ResetForm

```
public class ResetForm
```

Fields

PASSWORD

```
public static final String PASSWORD
```

PASSWORD_CONFIRMATION

```
public static final String PASSWORD_CONFIRMATION
```

Methods

getPassword

```
public String getPassword ()
```

getPasswordConfirmation

```
public String getPasswordConfirmation ()
```

getToken

```
public String getToken ()
```

setPassword

```
public void setPassword (String password)
```

setPasswordConfirmation

```
public void setPasswordConfirmation (String passwordConfirmation)
```

setToken

```
public void setToken (String token)
```

7.81.3 StartupForm

```
public class StartupForm
```

Fields

ADMIN_CONFIRM_PASSWORD

```
public static final String ADMIN_CONFIRM_PASSWORD
```

ADMIN_LOGIN

```
public static final String ADMIN_LOGIN
```

ADMIN_PASSWORD

```
public static final String ADMIN_PASSWORD
```

LANGUAGE

```
public static final String LANGUAGE
```

LOGIN_MODE

```
public static final String LOGIN_MODE
```

PROVIDER_NAME

```
public static final String PROVIDER_NAME
```

PROVIDER_URL

```
public static final String PROVIDER_URL
```

QUEUE_URL

```
public static final String QUEUE_URL
```

Methods

getAdminConfirmPassword

```
public String getAdminConfirmPassword ()
```

getAdminEmail

```
public String getAdminEmail ()
```

getAdminLogin

```
public String getAdminLogin ()
```

getAdminPassword

```
public String getAdminPassword ()
```

getLanguage

```
public String getLanguage ()
```

getLoginMode

```
public String getLoginMode ()
```

getProviderName

public **String** **getProviderName** ()

getProviderUrl

public **String** **getProviderUrl** ()

getQueueUrl

public **String** **getQueueUrl** ()

getSchedulerUrl

public **String** **getSchedulerUrl** ()

setAdminConfirmPassword

public void **setAdminConfirmPassword** (**String** *adminConfirmPassword*)

setAdminEmail

public void **setAdminEmail** (**String** *adminEmail*)

setAdminLogin

public void **setAdminLogin** (**String** *adminLogin*)

setAdminPassword

public void **setAdminPassword** (**String** *adminPassword*)

setLanguage

public void **setLanguage** (**String** *language*)

setLoginMode

public void **setLoginMode** (**String** *loginMode*)

setProviderName

public void **setProviderName** (**String** *providerName*)

setProviderUrl

```
public void setProviderUrl (String providerUrl)
```

setQueueUrl

```
public void setQueueUrl (String queueUrl)
```

setSchedulerUrl

```
public void setSchedulerUrl (String schedulerUrl)
```

7.81.4 StartupSuggestionsForm

```
public class StartupSuggestionsForm
```

Constructors**StartupSuggestionsForm**

```
public StartupSuggestionsForm ()
```

Methods**addDatabaseSuggestion**

```
public void addDatabaseSuggestion (String suggestion)
```

addQueueSuggestion

```
public void addQueueSuggestion (String suggestion)
```

addSchedulerSuggestion

```
public void addSchedulerSuggestion (String suggestion)
```

getDatabaseUrls

```
public List<String> getDatabaseUrls ()
```

getQueueUrls

```
public List<String> getQueueUrls ()
```

`getSchedulerUrls`

```
public List<String> getSchedulerUrls ()
```

7.81.5 UserInfo

```
public class UserInfo
```

Constructors

`UserInfo`

```
public UserInfo (String userName, boolean securityLaunch, String lang)
```

Methods

`equals`

```
public boolean equals (Object obj)
```

`getLang`

```
public String getLang ()
```

`getUserName`

```
public String getUserName ()
```

`hashCode`

```
public int hashCode ()
```

`isSecurityLaunch`

```
public boolean isSecurityLaunch ()
```

`toString`

```
public String toString ()
```

7.82 org.motechproject.server.web.helper

7.82.1 Header

public final class **Header**

Methods

generateHeader

public static *String* **generateHeader** (*Bundle bundle*)

7.82.2 Header.ElementOrder

public static class **ElementOrder**

Methods

getAfter

public *String* **getAfter** ()

getBefore

public *String* **getBefore** ()

getOrder

public *String* **getOrder** ()

getPath

public *String* **getPath** ()

setAfter

public void **setAfter** (*String after*)

setBefore

public void **setBefore** (*String before*)

setOrder

public void **setOrder** (*String order*)

setPath

```
public void setPath (String path)
```

7.82.3 Header.HeaderOrder

```
public static class HeaderOrder
```

Methods**getCss**

```
public List<ElementOrder> getCss ()
```

getJs

```
public List<ElementOrder> getJs ()
```

getLib

```
public List<ElementOrder> getLib ()
```

setCss

```
public void setCss (List<ElementOrder> css)
```

setJs

```
public void setJs (List<ElementOrder> js)
```

setLib

```
public void setLib (List<ElementOrder> lib)
```

7.82.4 MenuBuilder

```
public class MenuBuilder
```

Helper class for building the modules menu view(left-hand side nav). Modules to display are retrieved from the `UIFrameworkService`. Filters entries based on user permissions.

See also: `UIFrameworkService`

Methods

buildMenu

public ModuleMenu **buildMenu** (*String username*)

Builds the menu for the given user. Modules are retrieved from UIFrameworkService and filtered based on permissions.

Parameters

- **username** – username of the user for which the menu should be built.

Returns the built menu object.

7.82.5 SuggestionHelper

public class **SuggestionHelper**

Helper class for creating UI suggestions for the user. Checks default locations for available services.

Fields

DEFAULT_ACTIVEMQ_URL

public static final *String* **DEFAULT_ACTIVEMQ_URL**

Methods

suggestActivemqUrl

public *String* **suggestActivemqUrl** ()

Suggests the ActiveMQ url.

Returns The suggested url, or an empty string if the instance is not found.

7.83 org.motechproject.server.web.validator

7.83.1 AbstractValidator

public interface **AbstractValidator**

Basic interface which startup settings validators implement

Methods

validate

void **validate** (*StartupForm target*, *List<String> errors*, *ConfigSource configSource*)

7.83.2 OpenIdUserValidator

public class **OpenIdUserValidator** implements [AbstractValidator](#)
Validates presence of OpenId related field values Also validates provider URL

Constructors

OpenIdUserValidator

public **OpenIdUserValidator** ([UrlValidator](#) *urlValidator*)

Methods

validate

public void **validate** ([StartupForm](#) *target*, [List<String>](#) *errors*, [ConfigSource](#) *configSource*)

7.83.3 PersistedUserValidator

public class **PersistedUserValidator** implements [AbstractValidator](#)
Validates presence of admin user registration fields. Checks existence of user with identical name Checks existence of user with identical email Checks that password and confirmed password field are same.

Constructors

PersistedUserValidator

public **PersistedUserValidator** ([MotechUserService](#) *userService*)

Methods

validate

public void **validate** ([StartupForm](#) *target*, [List<String>](#) *errors*, [ConfigSource](#) *configSource*)

7.83.4 QueueURLValidator

public class **QueueURLValidator** implements [AbstractValidator](#)
Validates presence of Queue URL and if present whether it is in expected format or not

Constructors

QueueURLValidator

public **QueueURLValidator** ()

QueueURLValidator

```
public QueueURLValidator (UrlValidator urlValidator)
```

Methods

validate

```
public void validate (StartupForm target, List<String> errors, ConfigSource configSource)
```

7.83.5 RequiredFieldValidator

```
public class RequiredFieldValidator implements AbstractValidator  
    Generic validator class that validates presence of a given field
```

Fields

ERROR_REQUIRED

```
public static final String ERROR_REQUIRED
```

Constructors

RequiredFieldValidator

```
public RequiredFieldValidator (String fieldName, String fieldValue)
```

Methods

equals

```
public boolean equals (Object o)
```

hashCode

```
public int hashCode ()
```

validate

```
public void validate (StartupForm target, List<String> errors, ConfigSource configSource)
```

7.83.6 ResetFormValidator

```
public class ResetFormValidator
```

Methods

validate

```
public List<String> validate (ResetForm target)
```

7.83.7 StartupFormValidator

```
public class StartupFormValidator  
    StartupFormValidator validates user information during registration process
```

Constructors

StartupFormValidator

```
public StartupFormValidator ()
```

Methods

add

```
public void add (AbstractValidator validator)
```

getValidators

```
public List<AbstractValidator> getValidators ()
```

validate

```
public List<String> validate (StartupForm target, ConfigSource configSource)
```

7.83.8 StartupFormValidatorFactory

```
public class StartupFormValidatorFactory  
    Factory to create startUpFormValidator with requisite validators. If Admin User exists,the admin user is not  
    created so the relevant validators are not added.
```

Methods

getStartupFormValidator

```
public StartupFormValidator getStartupFormValidator (StartupForm startupSettings, MotechUserService userService)
```


7.83.9 UserRegistrationValidator

public class **UserRegistrationValidator** implements [AbstractValidator](#)

Validator to validate user registration details. Delegates to either [@OpenIdUserValidator](#) or [@UserRegistrationValidator](#) depending on login mode preference.

Constructors

UserRegistrationValidator

public **UserRegistrationValidator** ([PersistedUserValidator](#) *persistedUserValidator*, [OpenIdUserValidator](#) *openIdUserValidator*)

Methods

validate

public void **validate** ([StartupForm](#) *target*, [List<String>](#) *errors*, [ConfigSource](#) *configSource*)

7.83.10 ValidationUtils

public final class **ValidationUtils**

Validation utils that consists of common validations that can be used across multiple controllers.

Methods

validateEmptyOrWhitespace

public static void **validateEmptyOrWhitespace** ([Errors](#) *errors*, [String](#) *errorMessageFormat*, [String...](#) *fields*)

validateUrl

public static void **validateUrl** ([Errors](#) *errors*, [String](#) *dbUrlField*)

7.84 org.motechproject.tasks.annotations

7.84.1 TaskAction

public @interface **TaskAction**

7.84.2 TaskActionParam

public @interface **TaskActionParam**

Marks method parameter to be treated as action parameter.

Each parameter in the given method has to have this annotation otherwise it will be a problem with the proper execution of the channel action.

See also: `TaskAction`, `TaskChannel`, `TaskAnnotationBeanPostProcessor`

7.84.3 TaskAnnotationBeanPostProcessor

public class **TaskAnnotationBeanPostProcessor** implements `BeanPostProcessor`

Factory class which is looking for classes with `TaskChannel` annotation to add them to the channel definition as channel action.

See also: `TaskAction`, `TaskActionParam`, `TaskChannel`

Constructors

TaskAnnotationBeanPostProcessor

public **TaskAnnotationBeanPostProcessor** (`BundleContext` *bundleContext*, `ChannelService` *channelService*)

Methods

postProcessAfterInitialization

public `Object` **postProcessAfterInitialization** (`Object` *bean*, `String` *beanName*)

postProcessBeforeInitialization

public `Object` **postProcessBeforeInitialization** (`Object` *bean*, `String` *beanName*)

processAnnotations

public void **processAnnotations** (`ApplicationContext` *applicationContext*)

7.84.4 TaskChannel

public @interface **TaskChannel**

7.85 org.motechproject.tasks.contract

7.85.1 ActionEventRequest

public class **ActionEventRequest**

Constructors

ActionEventRequest

```
public ActionEventRequest (String displayName, String subject, String description, String serviceInterface, String serviceMethod, SortedSet<ActionParameterRequest> actionParameters)
```

ActionEventRequest

```
public ActionEventRequest (String displayName, String subject, String description, String serviceInterface, String serviceMethod)
```

Methods

addParameter

```
public void addParameter (ActionParameterRequest parameter, boolean changeOrder)
```

equals

```
public boolean equals (Object obj)
```

getActionParameters

```
public SortedSet<ActionParameterRequest> getActionParameters ()
```

getDescription

```
public String getDescription ()
```

getDisplayName

```
public String getDisplayName ()
```

getServiceInterface

```
public String getServiceInterface ()
```

getServiceMethod

```
public String getServiceMethod ()
```

getSubject

```
public String getSubject ()
```

hasService

```
public boolean hasService ()
```

hasSubject

```
public boolean hasSubject ()
```

hashCode

```
public int hashCode ()
```

isValid

```
public boolean isValid ()
```

toString

```
public String toString ()
```

7.85.2 ActionParameterRequest

```
public class ActionParameterRequest implements Comparable<ActionParameterRequest>  
    Object representation of a parameter in the channel action request definition.
```

See also: [ActionEventRequest](#)

Constructors

ActionParameterRequest

```
public ActionParameterRequest (String key, String displayName, Integer order, String type, boolean  
                                required)
```

ActionParameterRequest

```
public ActionParameterRequest (String key, String displayName, Integer order, String type)
```

ActionParameterRequest

```
public ActionParameterRequest (String key, String displayName, Integer order)
```

ActionParameterRequest

```
public ActionParameterRequest (String key, String displayName)
```

Methods

compareTo

public int **compareTo** ([ActionParameterRequest](#) *o*)

equals

public boolean **equals** ([Object](#) *obj*)

getDisplayName

public [String](#) **getDisplayName** ()

getKey

public [String](#) **getKey** ()

getOrder

public [Integer](#) **getOrder** ()

getType

public [String](#) **getType** ()

hashCode

public int **hashCode** ()

isRequired

public boolean **isRequired** ()

setOrder

public void **setOrder** (int *order*)

toString

public [String](#) **toString** ()

7.85.3 ChannelRequest

public class **ChannelRequest**
Service layer object denoting a `org.motechproject.tasks.domain.Channel`. Used by
`ChannelService`

Constructors

ChannelRequest

public **ChannelRequest** (*String displayName, String moduleName, String moduleVersion, String description, List<TriggerEventRequest> triggerTaskEvents, List<ActionEventRequest> actionTaskEvents*)

Methods

equals

public boolean **equals** (*Object obj*)

getActionTaskEvents

public List<ActionEventRequest> **getActionTaskEvents** ()

getDescription

public String **getDescription** ()

getDisplayName

public String **getDisplayName** ()

getModuleName

public String **getModuleName** ()

getModuleVersion

public String **getModuleVersion** ()

getTriggerTaskEvents

public List<TriggerEventRequest> **getTriggerTaskEvents** ()

hashCode

```
public int hashCode ()
```

setModuleName

```
public void setModuleName (String moduleName)
```

setModuleVersion

```
public void setModuleVersion (String moduleVersion)
```

toString

```
public String toString ()
```

7.85.4 EventParameterRequest

```
public class EventParameterRequest
```

Constructors**EventParameterRequest**

```
public EventParameterRequest (String displayName, String eventKey, String type)
```

EventParameterRequest

```
public EventParameterRequest (String displayName, String eventKey)
```

Methods**equals**

```
public boolean equals (Object obj)
```

getDisplayName

```
public String getDisplayName ()
```

getEventKey

```
public String getEventKey ()
```

getType

```
public String getType ()
```

hashCode

```
public int hashCode ()
```

toString

```
public String toString ()
```

7.85.5 TriggerEventRequest

```
public class TriggerEventRequest
```

Constructors**TriggerEventRequest**

```
public TriggerEventRequest (String displayName, String subject, String description,  
                           List<EventParameterRequest> eventParameters)
```

Methods**equals**

```
public boolean equals (Object obj)
```

getDescription

```
public String getDescription ()
```

getDisplayName

```
public String getDisplayName ()
```

getEventParameters

```
public List<EventParameterRequest> getEventParameters ()
```

getSubject

```
public String getSubject ()
```


hashCode

```
public int hashCode ()
```

toString

```
public String toString ()
```

7.86 org.motechproject.tasks.domain

7.86.1 ActionEvent

```
public class ActionEvent extends TaskEvent
```

Constructors

ActionEvent

```
public ActionEvent ()
```

ActionEvent

```
public ActionEvent (String displayName, String subject, String description, SortedSet<ActionParameter>  
                    actionParameters)
```

ActionEvent

```
public ActionEvent (String displayName, String description, String serviceInterface, String serviceMethod,  
                    SortedSet<ActionParameter> actionParameters)
```

ActionEvent

```
public ActionEvent (String displayName, String subject, String description, String serviceInterface, String  
                    serviceMethod, SortedSet<ActionParameter> actionParameters)
```

ActionEvent

```
public ActionEvent (ActionEventRequest actionEventRequest)
```

Methods

accept

```
public boolean accept (TaskActionInformation info)
```

addParameter

public void **addParameter** ([ActionParameter](#) *parameter*, boolean *changeOrder*)

containsParameter

public boolean **containsParameter** ([String](#) *key*)

equals

public boolean **equals** ([Object](#) *obj*)

getActionParameters

public [SortedSet](#)<[ActionParameter](#)> **getActionParameters** ()

getServiceInterface

public [String](#) **getServiceInterface** ()

getServiceMethod

public [String](#) **getServiceMethod** ()

hasService

public boolean **hasService** ()

hashCode

public int **hashCode** ()

setActionParameters

public void **setActionParameters** ([SortedSet](#)<[ActionParameter](#)> *actionParameters*)

setServiceInterface

public void **setServiceInterface** ([String](#) *serviceInterface*)

setServiceMethod

public void **setServiceMethod** ([String](#) *serviceMethod*)

toString

```
public String toString()
```

7.86.2 ActionParameter

public class **ActionParameter** extends [Parameter](#) implements [Comparable<ActionParameter>](#)
Object representation of a parameter in the channel action definition.

See also: [ActionEvent](#)

Constructors

ActionParameter

```
public ActionParameter()
```

ActionParameter

```
public ActionParameter(String displayName, String key)
```

ActionParameter

```
public ActionParameter(String displayName, String key, boolean required)
```

ActionParameter

```
public ActionParameter(String displayName, String key, Integer order)
```

ActionParameter

```
public ActionParameter(String displayName, String key, ParameterType type)
```

ActionParameter

```
public ActionParameter(String displayName, String key, ParameterType type, boolean required)
```

ActionParameter

```
public ActionParameter(String displayName, String key, ParameterType type, Integer order)
```

ActionParameter

```
public ActionParameter(ActionParameterRequest actionParameterRequest)
```

ActionParameter

```
public ActionParameter (String displayName, String key, ParameterType type, Integer order, boolean required)
```

Methods

compareTo

```
public int compareTo (ActionParameter o)
```

equals

```
public boolean equals (Object obj)
```

getKey

```
public String getKey ()
```

getOrder

```
public Integer getOrder ()
```

hashCode

```
public int hashCode ()
```

isRequired

```
public boolean isRequired ()
```

setKey

```
public void setKey (String key)
```

setOrder

```
public void setOrder (Integer order)
```

setRequired

```
public void setRequired (boolean required)
```

toString

```
public String toString ()
```

7.86.3 Channel

```
public class Channel
```

Constructors

Channel

```
public Channel ()
```

Channel

```
public Channel (String displayName, String moduleName, String moduleVersion)
```

Channel

```
public Channel (String displayName, String moduleName, String moduleVersion, String description,  
                List<TriggerEvent> triggerTaskEvents, List<ActionEvent> actionTaskEvents)
```

Channel

```
public Channel (ChannelRequest channelRequest)
```

Methods

addActionTaskEvent

```
public void addActionTaskEvent (ActionEvent actionEvent)
```

containsAction

```
public boolean containsAction (TaskActionInformation actionInformation)
```

containsTrigger

```
public boolean containsTrigger (TaskTriggerInformation triggerInformation)
```

equals

```
public boolean equals (Object obj)
```

getAction

```
public ActionEvent getAction (TaskActionInformation actionInformation)
```

getActionTaskEvents

```
public List<ActionEvent> getActionTaskEvents ()
```

getDescription

```
public String getDescription ()
```

getDisplayName

```
public String getDisplayName ()
```

getModuleName

```
public String getModuleName ()
```

getModuleVersion

```
public String getModuleVersion ()
```

getTrigger

```
public TriggerEvent getTrigger (TaskTriggerInformation triggerInformation)
```

getTriggerTaskEvents

```
public List<TriggerEvent> getTriggerTaskEvents ()
```

hashCode

```
public int hashCode ()
```

setActionTaskEvents

```
public void setActionTaskEvents (List<ActionEvent> actionTaskEvents)
```

setDescription

```
public void setDescription (String description)
```

setDisplayNames

```
public void setDisplayNames (String displayName)
```

setModuleName

```
public void setModuleName (String moduleName)
```

setModuleVersion

```
public void setModuleVersion (String moduleVersion)
```

setTriggerTaskEvents

```
public void setTriggerTaskEvents (List<TriggerEvent> triggerTaskEvents)
```

toString

```
public String toString ()
```

7.86.4 ChannelRegisterEvent

```
public class ChannelRegisterEvent
```

A wrapper over a `MotechEvent` with subject { @value EventSubjects#CHANNEL_REGISTER_SUBJECT }.
Raised when a channel is registered with the tasks module

Constructors**ChannelRegisterEvent**

```
public ChannelRegisterEvent (String moduleName)
```

ChannelRegisterEvent

```
public ChannelRegisterEvent (MotechEvent motechEvent)
```

Methods**getChannelModuleName**

```
public String getChannelModuleName ()
```

toMotechEvent

```
public MotechEvent toMotechEvent ()
```

7.86.5 DataSource

public class **DataSource** extends `TaskConfigStep`

Constructors

DataSource

public **DataSource** ()

DataSource

public **DataSource** (`Long providerId`, `Long objectId`, `String type`, `String name`, `List<Lookup> lookup`,
`boolean failIfDataNotFound`)

DataSource

public **DataSource** (`String providerName`, `Long providerId`, `Long objectId`, `String type`, `String name`,
`List<Lookup> lookup`, `boolean failIfDataNotFound`)

Methods

equals

public boolean **equals** (`Object obj`)

getLookup

public `List<Lookup>` **getLookup** ()

getName

public `String` **getName** ()

getObjectId

public `Long` **getObjectId** ()

getProviderId

public `Long` **getProviderId** ()

getProviderName

public `String` **getProviderName** ()

getType

```
public String getType ()
```

hashCode

```
public int hashCode ()
```

isFailIfDataNotFound

```
public boolean isFailIfDataNotFound ()
```

objectEquals

```
public boolean objectEquals (Long providerId, Long objectId, String type)
```

setFailIfDataNotFound

```
public void setFailIfDataNotFound (boolean failIfDataNotFound)
```

setLookup

```
public void setLookup (Object lookup)
```

setName

```
public void setName (String name)
```

setObjectId

```
public void setObjectId (Long objectId)
```

setProviderId

```
public void setProviderId (Long providerId)
```

setProviderName

```
public void setProviderName (String providerName)
```

setType

```
public void setType (String type)
```

toString

```
public String toString ()
```

7.86.6 EventParameter

```
public class EventParameter extends Parameter
```

Constructors

EventParameter

```
public EventParameter ()
```

EventParameter

```
public EventParameter (String displayName, String eventKey)
```

EventParameter

```
public EventParameter (String displayName, String eventKey, ParameterType type)
```

EventParameter

```
public EventParameter (EventParameterRequest eventParameterRequest)
```

Methods

equals

```
public boolean equals (Object obj)
```

getEventKey

```
public String getEventKey ()
```

hashCode

```
public int hashCode ()
```

setEventKey

```
public void setEventKey (String eventKey)
```

toString

```
public String toString ()
```

7.86.7 FieldParameter

```
public class FieldParameter extends Parameter
```

Constructors

FieldParameter

```
public FieldParameter ()
```

FieldParameter

```
public FieldParameter (String displayName, String fieldKey)
```

FieldParameter

```
public FieldParameter (String displayName, String fieldKey, ParameterType type)
```

Methods

equals

```
public boolean equals (Object obj)
```

getFieldKey

```
public String getFieldKey ()
```

hashCode

```
public int hashCode ()
```

setFieldKey

```
public void setFieldKey (String fieldKey)
```

toString

```
public String toString ()
```

7.86.8 Filter

public class **Filter** implements [Serializable](#)

Constructors

Filter

public **Filter** ()

Filter

public **Filter** ([EventParameter](#) *eventParameter*, boolean *negationOperator*, [String](#) *operator*, [String](#) *expression*)

Filter

public **Filter** ([String](#) *displayName*, [String](#) *key*, [ParameterType](#) *type*, boolean *negationOperator*, [String](#) *operator*, [String](#) *expression*)

Methods

equals

public boolean **equals** ([Object](#) *obj*)

getDisplayName

public [String](#) **getDisplayName** ()

getExpression

public [String](#) **getExpression** ()

getKey

public [String](#) **getKey** ()

getOperator

public [String](#) **getOperator** ()

getType

public [ParameterType](#) **getType** ()

hashCode

```
public int hashCode ()
```

isNegationOperator

```
public boolean isNegationOperator ()
```

setDisplayName

```
public void setDisplayName (String displayName)
```

setExpression

```
public void setExpression (String expression)
```

setKey

```
public void setKey (String key)
```

setNegationOperator

```
public void setNegationOperator (boolean negationOperator)
```

setOperator

```
public void setOperator (String operator)
```

setType

```
public void setType (ParameterType type)
```

toString

```
public String toString ()
```

7.86.9 FilterSet

```
public class FilterSet extends TaskConfigStep
```

Constructors**FilterSet**

```
public FilterSet ()
```

FilterSet

```
public FilterSet (List<Filter> filters)
```

Methods

addFilter

```
public void addFilter (Filter filter)
```

equals

```
public boolean equals (Object obj)
```

getFilters

```
public List<Filter> getFilters ()
```

hashCode

```
public int hashCode ()
```

setFilters

```
public void setFilters (List<Filter> filters)
```

toString

```
public String toString ()
```

7.86.10 KeyInformation

```
public final class KeyInformation
```

Object representation of dragged field from trigger or data source.

This class represents a single dragged field from trigger or data source. This class does not expose a public constructor. You have to use `parse(String)` method if you want to parse single field or `parseAll(String)` if you want to get all fields from a given string.

Fields

ADDITIONAL_DATA_PREFIX

```
public static final String ADDITIONAL_DATA_PREFIX
```

Prefix which is used for data source fields.

TRIGGER_PREFIX

public static final `String` **TRIGGER_PREFIX**
Prefix which is used for trigger fields.

Methods

equals

public boolean **equals** (`Object obj`)

fromAdditionalData

public boolean **fromAdditionalData** ()
Check if the field is from the data source.
Returns true if the field is from the data source otherwise false

fromTrigger

public boolean **fromTrigger** ()
Check if the field is from the trigger.
Returns true if the field is from the trigger otherwise false

getDataProviderId

public `Long` **getDataProviderId** ()

getKey

public `String` **getKey** ()

getManipulations

public `List<String>` **getManipulations** ()
Get manipulations assigned to the field.
Returns list of manipulations

getObjectId

public `Long` **getObjectId** ()

getObjectType

public `String` **getObjectType** ()

getOriginalKey

```
public String getOriginalKey ()
```

Get original representation of the dragged field.

Returns string representation of the field

getPrefix

```
public String getPrefix ()
```

hasManipulations

```
public boolean hasManipulations ()
```

Check if the field has any manipulations.

Returns true if the field has manipulations otherwise false

hashCode

```
public int hashCode ()
```

parse

```
public static KeyInformation parse (String input)
```

Parse given string to instance of `KeyInformation`.

This method should be used to convert string representation of dragged field to instance of `KeyInformation`.

Argument has adhere to one of the following format:

- trigger field format: **trigger.eventKey**
- data source format: **ad.dataProviderId.objectType#objectId.fieldKey**

Argument can also contain list of manipulation which should be executed on field before it will be used by `org.motechproject.tasks.service.TaskTriggerHandler` class. Manipulations should be connected together by the `?` character.

Example of input argument:

- ad.279f5fdf60700d9717270b1ae3011eb1.CaseInfo#0.fieldValues.phu_id
- trigger.message?format(Ala,cat)?capitalize

Parameters

- **input** – string representation of a dragged field from trigger or data source

Throws

- **IllegalArgumentException** – exception is thrown if format for data source field is incorrect or if the dragged field is not from trigger or data source.

Returns Object representation of a dragged field

parseAll

public static List<KeyInformation> **parseAll** (String input)

Find all fields from given input and convert them to the instance of *KeyInformation*.

This method should be used to find and convert all of string representation of the field from trigger and/or data sources. Fields in input have to adhere to one of the following formats:

- trigger field format: `{{trigger.eventKey}}`
- data source format: `{{ad.dataProviderId.objectType#objectId.fieldKey}}`

To find fields in the input argument this method uses regular expression. When field is found it is converted to an instance of *KeyInformation* by using the `parse(String)` method.

Fields are found by the following regular expression: `{{((.*?))}}(?![^(]*))`. The expression searches for strings that start with `{{` and end with `}}` and are not within `(` and `)`. Because of manipulations which contain additional data in `(...)` needed to execute manipulation on the field (e.g.: join needs to have the join character) and the text in `(...)` can be another string representation of the dragged field, the expression has to check if the field has this kind of manipulation.

Example of input argument:

- `{{trigger.message?format(Ala,cat)?capitalize}}`
- You get the following message: `{{trigger.message}}`

Parameters

- **input** – string with one or more string representation of dragged fields from trigger and/or data sources

Throws

- **IllegalArgumentException** – in the same situations as the `parse(String)` method.

Returns list of object representation of dragged fields.

7.86.11 Lookup

public class **Lookup** implements *Serializable*

Constructors

Lookup

public **Lookup** ()

Lookup

public **Lookup** (String field, String value)

Methods

equals

public boolean **equals** (*Object obj*)

getField

public *String* **getField** ()

getValue

public *String* **getValue** ()

hashCode

public int **hashCode** ()

setField

public void **setField** (*String field*)

setValue

public void **setValue** (*String value*)

7.86.12 LookupFieldsParameter

public class **LookupFieldsParameter**

Constructors

LookupFieldsParameter

public **LookupFieldsParameter** ()

LookupFieldsParameter

public **LookupFieldsParameter** (*String displayName*, *List<String> fields*)

Methods

equals

public boolean **equals** (*Object obj*)

getDisplayName

```
public String getDisplayName ()
```

getFields

```
public List<String> getFields ()
```

hashCode

```
public int hashCode ()
```

setDisplayName

```
public void setDisplayName (String displayName)
```

setFields

```
public void setFields (List<String> fields)
```

toString

```
public String toString ()
```

7.86.13 ManipulationTarget

```
public enum ManipulationTarget
```

Defines the target of various manipulations used in a task for both triggers and data sources.

Enum Constants**ALL**

```
public static final ManipulationTarget ALL
```

DATE

```
public static final ManipulationTarget DATE
```

STRING

```
public static final ManipulationTarget STRING
```

7.86.14 ManipulationType

public enum **ManipulationType**

Defines the type of various manipulations used in a task for both triggers and data sources.

Enum Constants

CAPITALIZE

public static final ManipulationType **CAPITALIZE**

DATETIME

public static final ManipulationType **DATETIME**

FORMAT

public static final ManipulationType **FORMAT**

JOIN

public static final ManipulationType **JOIN**

PLUSDAYS

public static final ManipulationType **PLUSDAYS**

SPLIT

public static final ManipulationType **SPLIT**

SUBSTRING

public static final ManipulationType **SUBSTRING**

TOLOWER

public static final ManipulationType **TOLOWER**

TOUPPER

public static final ManipulationType **TOUPPER**

UNKNOWN

public static final [ManipulationType](#) UNKNOWN

7.86.15 OperatorType

public enum **OperatorType**

Object representation of available operators in filter definition.

Enum Constants

AFTER

public static final [OperatorType](#) AFTER

AFTER_NOW

public static final [OperatorType](#) AFTER_NOW

BEFORE

public static final [OperatorType](#) BEFORE

BEFORE_NOW

public static final [OperatorType](#) BEFORE_NOW

CONTAINS

public static final [OperatorType](#) CONTAINS

ENDSWITH

public static final [OperatorType](#) ENDSWITH

EQUALS

public static final [OperatorType](#) EQUALS

EQ_NUMBER

public static final [OperatorType](#) EQ_NUMBER

EXIST

public static final [OperatorType](#) **EXIST**

GT

public static final [OperatorType](#) **GT**

LESS_DAYS_FROM_NOW

public static final [OperatorType](#) **LESS_DAYS_FROM_NOW**

LESS_MONTHS_FROM_NOW

public static final [OperatorType](#) **LESS_MONTHS_FROM_NOW**

LT

public static final [OperatorType](#) **LT**

MORE_DAYS_FROM_NOW

public static final [OperatorType](#) **MORE_DAYS_FROM_NOW**

MORE_MONTHS_FROM_NOW

public static final [OperatorType](#) **MORE_MONTHS_FROM_NOW**

STARTSWITH

public static final [OperatorType](#) **STARTSWITH**

7.86.16 Parameter

public abstract class **Parameter** implements [Serializable](#)

Constructors

Parameter

protected **Parameter** ()

Parameter

protected **Parameter** (*String displayName*, *ParameterType type*)

Methods

equals

public boolean **equals** (*Object obj*)

getDisplayName

public *String* **getDisplayName** ()

getType

public *ParameterType* **getType** ()

hashCode

public int **hashCode** ()

setDisplayName

public void **setDisplayName** (*String displayName*)

setType

public void **setType** (*ParameterType type*)

toString

public *String* **toString** ()

7.86.17 ParameterType

public enum **ParameterType**

Defines the type of various values used in a task including trigger parameters, action parameters and data source object fields.

Enum Constants

BOOLEAN

public static final *ParameterType* **BOOLEAN**

DATE

public static final *ParameterType* **DATE**

DOUBLE

public static final `ParameterType` **DOUBLE**

INTEGER

public static final `ParameterType` **INTEGER**

LIST

public static final `ParameterType` **LIST**

LONG

public static final `ParameterType` **LONG**

MAP

public static final `ParameterType` **MAP**

TEXTAREA

public static final `ParameterType` **TEXTAREA**

TIME

public static final `ParameterType` **TIME**

UNICODE

public static final `ParameterType` **UNICODE**

UNKNOWN

public static final `ParameterType` **UNKNOWN**

7.86.18 SettingsDto

public class **SettingsDto**

Methods

`getTaskPossibleErrors`

public `String` `getTaskPossibleErrors` ()

isValid

```
public boolean isValid()
```

setTaskPossibleErrors

```
public void setTaskPossibleErrors (String taskPossibleErrors)
```

7.86.19 Task

```
public class Task
```

A task is set of actions that are executed in response to a trigger. The actions and the trigger are defined by their respective `Channels`.

Constructors

Task

```
public Task ()
```

Task

```
public Task (String name, TaskTriggerInformation trigger, List<TaskActionInformation> actions)
```

Task

```
public Task (String name, TaskTriggerInformation trigger, List<TaskActionInformation> actions, TaskConfig taskConfig, boolean enabled, boolean hasRegisteredChannel)
```

Methods

addAction

```
public void addAction (TaskActionInformation action)
```

addValidationErrors

```
public void addValidationErrors (Set<TaskError> validationErrors)
```

equals

```
public boolean equals (Object obj)
```

getActions

```
public List<TaskActionInformation> getActions ()
```

getDescription

```
public String getDescription ()
```

getId

```
public Long getId ()
```

getName

```
public String getName ()
```

getTaskConfig

```
public TaskConfig getTaskConfig ()
```

getTrigger

```
public TaskTriggerInformation getTrigger ()
```

getValidationErrors

```
public Set<TaskError> getValidationErrors ()
```

hasRegisteredChannel

```
public boolean hasRegisteredChannel ()
```

hasValidationErrors

```
public boolean hasValidationErrors ()
```

hashCode

```
public int hashCode ()
```

isEnabled

```
public boolean isEnabled ()
```

removeValidationError

public void **removeValidationError** (*String* *message*)

setActions

public void **setActions** (*List*<*TaskActionInformation*> *actions*)

setDescription

public void **setDescription** (*String* *description*)

setEnabled

public void **setEnabled** (boolean *enabled*)

setHasRegisteredChannel

public void **setHasRegisteredChannel** (boolean *hasRegisteredChannel*)

setId

public void **setId** (*Long* *id*)

setName

public void **setName** (*String* *name*)

setTaskConfig

public void **setTaskConfig** (*TaskConfig* *taskConfig*)

setTrigger

public void **setTrigger** (*TaskTriggerInformation* *trigger*)

setValidationErrors

public void **setValidationErrors** (*Set*<*TaskError*> *validationErrors*)

toString

public *String* **toString** ()

7.86.20 TaskActionInformation

public class **TaskActionInformation** extends [TaskEventInformation](#)
Represents an action event configured with a [Task](#)

Constructors

TaskActionInformation

```
public TaskActionInformation ()
```

TaskActionInformation

```
public TaskActionInformation (String displayName, String channelName, String moduleName, String  
moduleVersion, String subject)
```

TaskActionInformation

```
public TaskActionInformation (String displayName, String channelName, String moduleName, String  
moduleVersion, String subject, Map<String, String> values)
```

TaskActionInformation

```
public TaskActionInformation (String displayName, String channelName, String moduleName, String  
moduleVersion, String serviceInterface, String serviceMethod)
```

TaskActionInformation

```
public TaskActionInformation (String displayName, String channelName, String moduleName, String  
moduleVersion, String subject, String serviceInterface, String serviceMethod, Map<String, String> values)
```

Methods

equals

```
public boolean equals (Object obj)
```

getServiceInterface

```
public String getServiceInterface ()
```

getServiceMethod

```
public String getServiceMethod ()
```

getValues

```
public Map<String, String> getValues ()
```

hasService

```
public boolean hasService ()
```

hashCode

```
public int hashCode ()
```

setServiceInterface

```
public void setServiceInterface (String serviceInterface)
```

setServiceMethod

```
public void setServiceMethod (String serviceMethod)
```

setValues

```
public void setValues (Map<String, String> values)
```

toString

```
public String toString ()
```

7.86.21 TaskActivity

```
public class TaskActivity implements Comparable<TaskActivity>
```

Constructors**TaskActivity**

```
public TaskActivity ()
```

TaskActivity

```
public TaskActivity (String message, Long task, TaskActivityType activityType)
```

TaskActivity

```
public TaskActivity (String message, String field, Long task, TaskActivityType activityType)
```

TaskActivity

```
public TaskActivity (String message, List<String> fields, Long task, TaskActivityType activityType)
```

TaskActivity

```
public TaskActivity (String message, List<String> fields, Long task, TaskActivityType activityType, String stackTraceElement)
```

Methods

compareTo

```
public int compareTo (TaskActivity o)
```

equals

```
public boolean equals (Object obj)
```

getActivityType

```
public TaskActivityType getActivityType ()
```

getDate

```
public DateTime getDate ()
```

getFields

```
public List<String> getFields ()
```

getMessage

```
public String getMessage ()
```

getStackTraceElement

```
public String getStackTraceElement ()
```

getTask

```
public Long getTask ()
```

hashCode

```
public int hashCode ()
```

setActivityType

```
public void setActivityType (TaskActivityType activityType)
```

setDate

```
public void setDate (DateTime date)
```

setField

```
public void setField (String field)
```

setFields

```
public void setFields (List<String> fields)
```

setMessage

```
public void setMessage (String message)
```

setStackTraceElement

```
public void setStackTraceElement (String stackTraceElement)
```

setTask

```
public void setTask (Long task)
```

toString

```
public String toString ()
```

7.86.22 TaskActivityType

```
public enum TaskActivityType
```

Enum Constants**ERROR**

```
public static final TaskActivityType ERROR
```

SUCCESS

public static final `TaskActivityType` **SUCCESS**

WARNING

public static final `TaskActivityType` **WARNING**

7.86.23 TaskBuilder

public class **TaskBuilder**

Constructors

TaskBuilder

public **TaskBuilder** ()

Methods

addAction

public `TaskBuilder` **addAction** (`TaskActionInformation` *action*)

addDataSource

public `TaskBuilder` **addDataSource** (`DataSource` *dataSource*)

addFilterSet

public `TaskBuilder` **addFilterSet** (`FilterSet` *filterSet*)

build

public `Task` **build** ()

clear

public `TaskBuilder` **clear** ()

isEnabled

public `TaskBuilder` **isEnabled** (boolean *enabled*)

withDescription

```
public TaskBuilder withDescription (String description)
```

withId

```
public TaskBuilder withId (Long id)
```

withName

```
public TaskBuilder withName (String name)
```

withTaskConfig

```
public TaskBuilder withTaskConfig (TaskConfig taskConfig)
```

withTrigger

```
public TaskBuilder withTrigger (TaskTriggerInformation trigger)
```

7.86.24 TaskConfig

```
public class TaskConfig implements Serializable
```

Methods**add**

```
public TaskConfig add (TaskConfigStep... configSteps)
```

addAll

```
public TaskConfig addAll (SortedSet<TaskConfigStep> set)
```

equals

```
public boolean equals (Object obj)
```

getDataSource

```
public DataSource getDataSource (Long providerId, Long objectId, String objectType)
```

getDataSources

```
public List<DataSource> getDataSources ()
```

getDataSources

public [SortedSet](#)<[DataSource](#)> **getDataSources** ([Long](#) *providerId*)

getFilters

public [List](#)<[FilterSet](#)> **getFilters** ()

getSteps

public [SortedSet](#)<[TaskConfigStep](#)> **getSteps** ()

hashCode

public int **hashCode** ()

removeAll

public [TaskConfig](#) **removeAll** ()

removeDataSources

public [TaskConfig](#) **removeDataSources** ()

removeFilterSets

public [TaskConfig](#) **removeFilterSets** ()

setDataSources

public void **setDataSources** ([List](#)<[DataSource](#)> *dataSources*)

setFilters

public void **setFilters** ([List](#)<[FilterSet](#)> *filters*)

toString

public [String](#) **toString** ()

7.86.25 TaskConfigStep

public abstract class **TaskConfigStep** implements [Comparable](#)<[TaskConfigStep](#)>, [Serializable](#)

Methods

`compareTo`

public int **compareTo** (*TaskConfigStep o*)

`equals`

public boolean **equals** (*Object obj*)

`getOrder`

public *Integer* **getOrder** ()

`hashCode`

public int **hashCode** ()

`setOrder`

public void **setOrder** (*Integer order*)

`toString`

public *String* **toString** ()

7.86.26 TaskDataProvider

public class **TaskDataProvider**

The `TaskDataProvider` class contains all informations about data providers used in tasks

Constructors

`TaskDataProvider`

public **TaskDataProvider** ()

`TaskDataProvider`

public **TaskDataProvider** (*String name*, *List<TaskDataProviderObject> objects*)

Methods

containsProviderObject

public boolean **containsProviderObject** (*String type*)

containsProviderObjectField

public boolean **containsProviderObjectField** (*String type*, *String fieldKey*)

containsProviderObjectLookup

public boolean **containsProviderObjectLookup** (*String type*, *String lookupField*)

equals

public boolean **equals** (*Object obj*)

getId

public *Long* **getId** ()

getKeyType

public *String* **getKeyType** (*String key*)

getName

public *String* **getName** ()

getObjects

public *List*<*TaskDataProviderObject*> **getObjects** ()

getProviderObject

public *TaskDataProviderObject* **getProviderObject** (*String type*)

hashCode

public int **hashCode** ()

setId

public void **setId** (*Long id*)

setName

```
public void setName (String name)
```

setObjects

```
public void setObjects (List<TaskDataProviderObject> objects)
```

toString

```
public String toString ()
```

7.86.27 TaskDataProviderObject

```
public class TaskDataProviderObject implements Serializable
```

Constructors**TaskDataProviderObject**

```
public TaskDataProviderObject ()
```

TaskDataProviderObject

```
public TaskDataProviderObject (String displayName, String type, List<LookupFieldsParameter>  
                                lookupFields, List<FieldParameter> fields)
```

Methods**containsField**

```
public boolean containsField (String fieldKey)
```

equals

```
public boolean equals (Object obj)
```

getDisplayName

```
public String getDisplayName ()
```

getFields

```
public List<FieldParameter> getFields ()
```

getLookupFields

```
public List<LookupFieldsParameter> getLookupFields ()
```

getType

```
public String getType ()
```

hashCode

```
public int hashCode ()
```

setDisplayName

```
public void setDisplayName (String displayName)
```

setFields

```
public void setFields (List<FieldParameter> fields)
```

setLookupFields

```
public void setLookupFields (List<Object> lookupFields)
```

setType

```
public void setType (String type)
```

toString

```
public String toString ()
```

7.86.28 TaskError

```
public class TaskError implements Serializable
```

Constructors

TaskError

```
public TaskError ()
```

TaskError

```
public TaskError (TaskErrorType type, String... args)
```

TaskError

public **TaskError** (*String message*, *String... args*)

Methods

equals

public boolean **equals** (*Object obj*)

getArgs

public *List<String>* **getArgs** ()

getMessage

public *String* **getMessage** ()

hashCode

public int **hashCode** ()

setArgs

public void **setArgs** (*List<String> args*)

setMessage

public void **setMessage** (*String message*)

toString

public *String* **toString** ()

7.86.29 TaskErrorType

public enum **TaskErrorType**

Enum Constants

BLANK

public static final *TaskErrorType* **BLANK**

EMPTY_COLLECTION

public static final [TaskErrorType](#) **EMPTY_COLLECTION**

NULL

public static final [TaskErrorType](#) **NULL**

VERSION

public static final [TaskErrorType](#) **VERSION**

7.86.30 TaskEvent

public abstract class **TaskEvent** implements [Serializable](#)

Constructors

TaskEvent

protected **TaskEvent** ()

TaskEvent

protected **TaskEvent** ([String](#) *description*, [String](#) *displayName*, [String](#) *subject*)

Methods

containsParameter

public boolean **containsParameter** ([String](#) *key*)

equals

public boolean **equals** ([Object](#) *obj*)

equalsSubject

protected boolean **equalsSubject** ([String](#) *subject*)

getDescription

public [String](#) **getDescription** ()

getDisplayName

```
public String getDisplayName ()
```

getSubject

```
public String getSubject ()
```

hasSubject

```
public boolean hasSubject ()
```

hashCode

```
public int hashCode ()
```

setDescription

```
public void setDescription (String description)
```

setDisplayName

```
public void setDisplayName (String displayName)
```

setSubject

```
public void setSubject (String subject)
```

toString

```
public String toString ()
```

7.86.31 TaskEventInformation

```
public abstract class TaskEventInformation implements Serializable
```

Constructors**TaskEventInformation**

```
public TaskEventInformation ()
```

TaskEventInformation

```
public TaskEventInformation (String displayName, String channelName, String moduleName, String
                               moduleVersion, String subject)
```

Methods

equals

```
public boolean equals (Object obj)
```

getChannelName

```
public String getChannelName ()
```

getDisplayName

```
public String getDisplayName ()
```

getModuleName

```
public String getModuleName ()
```

getModuleVersion

```
public String getModuleVersion ()
```

getSubject

```
public String getSubject ()
```

hasSubject

```
public boolean hasSubject ()
```

hashCode

```
public int hashCode ()
```

setChannelName

```
public void setChannelName (String channelName)
```

setDisplayNames

```
public void setDisplayNames (String displayName)
```

setModuleName

```
public void setModuleName (String moduleName)
```

setModuleVersion

```
public void setModuleVersion (String moduleVersion)
```

setSubject

```
public void setSubject (String subject)
```

toString

```
public String toString ()
```

7.86.32 TaskTriggerInformation

```
public class TaskTriggerInformation extends TaskEventInformation
```

Constructors

TaskTriggerInformation

```
public TaskTriggerInformation ()
```

TaskTriggerInformation

```
public TaskTriggerInformation (String displayName, String channelName, String moduleName, String moduleVersion, String subject)
```

TaskTriggerInformation

```
public TaskTriggerInformation (TaskTriggerInformation other)
```

7.86.33 TriggerEvent

```
public class TriggerEvent extends TaskEvent
```

The `TriggerEvent` class is responsible for storing data about event

Constructors

TriggerEvent

```
public TriggerEvent ()
```

TriggerEvent

```
public TriggerEvent (String displayName, String subject, String description, List<EventParameter> event-  
Parameters)
```

TriggerEvent

```
public TriggerEvent (TriggerEventRequest triggerEventRequest)
```

Methods

containsParameter

```
public boolean containsParameter (String key)
```

equals

```
public boolean equals (Object obj)
```

getEventParameters

```
public List<EventParameter> getEventParameters ()
```

getKeyType

```
public String getKeyType (String key)
```

hashCode

```
public int hashCode ()
```

setEventParameters

```
public void setEventParameters (List<EventParameter> eventParameters)
```

toString

```
public String toString ()
```

7.87 org.motechproject.tasks.ex

7.87.1 ActionNotFoundException

public class **ActionNotFoundException** extends [Exception](#)

Constructors

ActionNotFoundException

public **ActionNotFoundException** ([String](#) *message*)

7.87.2 TaskHandlerException

public class **TaskHandlerException** extends [Exception](#)

Constructors

TaskHandlerException

public **TaskHandlerException** ([TaskFailureCause](#) *failureCause*, [String](#) *messageKey*)

TaskHandlerException

public **TaskHandlerException** ([TaskFailureCause](#) *failureCause*, [String](#) *messageKey*, [String](#)... *args*)

TaskHandlerException

public **TaskHandlerException** ([TaskFailureCause](#) *failureCause*, [String](#) *messageKey*, [Throwable](#) *cause*,
[String](#)... *args*)

Methods

getArgs

public [List](#)<[String](#)> **getArgs** ()

getFailureCause

public [TaskFailureCause](#) **getFailureCause** ()

getMessage

public [String](#) **getMessage** ()

7.87.3 TaskNotFoundException

public class **TaskNotFoundException** extends [IllegalArgumentException](#)

Constructors

TaskNotFoundException

public **TaskNotFoundException** ([Long](#) *taskId*)

7.87.4 TriggerNotFoundException

public class **TriggerNotFoundException** extends [Exception](#)

Constructors

TriggerNotFoundException

public **TriggerNotFoundException** ([String](#) *message*)

7.87.5 ValidationException

public class **ValidationException** extends [IllegalArgumentException](#)

Constructors

ValidationException

public **ValidationException** ([String](#) *objectType*, [Set](#)<[TaskError](#)> *taskErrors*)

Methods

getMessage

public [String](#) **getMessage** ()

getTaskErrors

public [Set](#)<[TaskError](#)> **getTaskErrors** ()

7.88 org.motechproject.tasks.json

7.88.1 ActionEventRequestDeserializer

public class **ActionEventRequestDeserializer** implements [JsonDeserializer](#)<[ActionEventRequest](#)>

Fields

ACTION_PARAMETERS_FIELD

```
public static final String ACTION_PARAMETERS_FIELD
```

DESCRIPTION_FIELD

```
public static final String DESCRIPTION_FIELD
```

DISPLAY_NAME_FIELD

```
public static final String DISPLAY_NAME_FIELD
```

SERVICE_INTERFACE_FIELD

```
public static final String SERVICE_INTERFACE_FIELD
```

SERVICE_METHOD_FIELD

```
public static final String SERVICE_METHOD_FIELD
```

SUBJECT_FIELD

```
public static final String SUBJECT_FIELD
```

Methods

deserialize

```
public ActionEventRequest deserialize (JsonElement element, Type type, JsonDeserializationContext  
                                         context)
```

7.88.2 TaskConfigDeserializer

```
public class TaskConfigDeserializer extends JsonSerializer<TaskConfig>
```

Methods

deserialize

```
public TaskConfig deserialize (JsonParser parser, DeserializationContext context)
```

7.88.3 TaskDeserializer

public class **TaskDeserializer** extends JsonSerializer<Task>

Methods

deserialize

public Task **deserialize** (JsonParser *jsonParser*, DeserializationContext *deserializationContext*)

7.89 org.motechproject.tasks.repository

7.89.1 ChannelsDataService

public interface **ChannelsDataService** extends MotechDataService<Channel>

Methods

findByModuleName

Channel **findByModuleName** (String *moduleName*)

7.89.2 TaskActivitiesDataService

public interface **TaskActivitiesDataService** extends MotechDataService<TaskActivity>

Methods

byTask

List<TaskActivity> **byTask** (Long *task*)

7.89.3 TasksDataService

public interface **TasksDataService** extends MotechDataService<Task>

7.90 org.motechproject.tasks.service

7.90.1 ChannelService

public interface **ChannelService**

Manages CRUD operations for a Channel.

Methods

addOrUpdate

void **addOrUpdate** ([Channel](#) *channel*)

getAllChannels

[List<Channel>](#) **getAllChannels** ()

getChannel

[Channel](#) **getChannel** ([String](#) *moduleName*)

getChannelIcon

[BundleIcon](#) **getChannelIcon** ([String](#) *moduleName*)

registerChannel

void **registerChannel** ([ChannelRequest](#) *channelRequest*)

registerChannel

void **registerChannel** ([InputStream](#) *stream*, [String](#) *moduleName*, [String](#) *moduleVersion*)

7.90.2 DataProviderDataService

public interface **DataProviderDataService** extends [MotechDataService<TaskDataProvider>](#)

Methods

findByName

[TaskDataProvider](#) **findByName** ([String](#) *name*)

7.90.3 DataSourceObject

public class **DataSourceObject**

DataSourceObject is the result of a `org.motechproject.commons.api.DataProvider` lookup.

Constructors

DataSourceObject

```
public DataSourceObject (String objectId, Object objectValue, boolean failIfNotFound)
```

Methods

equals

```
public boolean equals (Object o)
```

getObjectId

```
public String getObjectId ()
```

getObjectValue

```
public Object getObjectValue ()
```

hashCode

```
public int hashCode ()
```

isFailIfNotFound

```
public boolean isFailIfNotFound ()
```

7.90.4 HandlerPredicates

final class **HandlerPredicates**

Utility class defining filters over some collections.

Methods

activeTasks

```
public static Predicate activeTasks ()
```

withServiceName

```
public static Predicate withServiceName (String serviceName)
```

7.90.5 KeyEvaluator

public class **KeyEvaluator**

KeyEvaluator evaluates the value of a key in the context of a task which is used to execute filters and actions.

Constructors

KeyEvaluator

public **KeyEvaluator** ([TaskContext](#) *taskContext*)

Methods

evaluateTemplateString

public [String](#) **evaluateTemplateString** ([String](#) *template*)

getValue

public [Object](#) **getValue** ([KeyInformation](#) *keyInformation*)

manipulate

[String](#) **manipulate** ([String](#) *manipulation*, [String](#) *value*)

7.90.6 MethodHandler

class **MethodHandler**

Utility class used by [TaskTriggerHandler](#) to construct a list of parameter types of the method in the correct order.

See also: [TaskTriggerHandler](#)

Constructors

MethodHandler

public **MethodHandler** ([ActionEvent](#) *action*, [Map](#)<[String](#), [Object](#)> *parameters*)

Methods

getClasses

public [Class](#)[] **getClasses** ()

getObjects

```
public Object[] getObjects ()
```

isParametrized

```
public boolean isParametrized ()
```

7.90.7 TaskActionExecutor

```
class TaskActionExecutor
```

Builds action parameters from `TaskContext` and executes the action by invoking its service or raising its event.

Constructors

TaskActionExecutor

```
TaskActionExecutor (TaskService taskService, TaskActivityService activityService, EventRelay eventRelay)
```

Methods

createParameters

```
Map<String, Object> createParameters (TaskActionInformation info, ActionEvent action)
```

execute

```
void execute (Task task, TaskActionInformation actionInformation, TaskContext taskContext)
```

setBundleContext

```
void setBundleContext (BundleContext bundleContext)
```

7.90.8 TaskActivityService

```
public interface TaskActivityService
```

Methods

addError

```
void addError (Task task, TaskHandlerException e)
```

addSuccess

void **addSuccess** ([Task](#) *task*)

addWarning

void **addWarning** ([Task](#) *task*)

addWarning

void **addWarning** ([Task](#) *task*, [String](#) *key*, [String](#) *value*)

addWarning

void **addWarning** ([Task](#) *task*, [String](#) *key*, [String](#) *field*, [Exception](#) *e*)

deleteActivitiesForTask

void **deleteActivitiesForTask** ([Long](#) *taskId*)

errorsFromLastRun

[List](#)<[TaskActivity](#)> **errorsFromLastRun** ([Task](#) *task*)

getAllActivities

[List](#)<[TaskActivity](#)> **getAllActivities** ()

getTaskActivities

[List](#)<[TaskActivity](#)> **getTaskActivities** ([Long](#) *taskId*)

7.90.9 TaskContext

public class **TaskContext**

TaskContext holds task trigger event and data provider lookup objects that are used while executing filters/actions.

Constructors**TaskContext**

public **TaskContext** ([Task](#) *task*, [MotechEvent](#) *event*, [TaskActivityService](#) *activityService*)

Methods

addDataSourceObject

```
public void addDataSourceObject (String objectId, Object dataSourceObject, boolean failIfDataNot-Found)
```

getDataSourceObjectValue

```
public Object getDataSourceObjectValue (String objectId, String field, String objectType)
```

getTask

```
public Task getTask ()
```

getTriggerParameters

```
public Map<String, Object> getTriggerParameters ()
```

getTriggerValue

```
public Object getTriggerValue (String key)
```

publishWarningActivity

```
public void publishWarningActivity (String message, String field)
```

7.90.10 TaskDataProviderService

```
public interface TaskDataProviderService
```

Methods

getProvider

```
TaskDataProvider getProvider (String name)
```

getProviderById

```
TaskDataProvider getProviderById (Long providerId)
```

getProviders

```
List<TaskDataProvider> getProviders ()
```

registerProvider

void **registerProvider** ([String](#) *json*)

registerProvider

void **registerProvider** ([InputStream](#) *stream*)

7.90.11 TaskFilterExecutor

public class **TaskFilterExecutor**

The `TaskFilterExecutor` applies a list of filters in a `#TaskContext`.

- convertTo** - convert a given value to a correct type,
- getFieldValue** - get value of a field defined in the key from the given object,
- getTriggerKey** - get value of a trigger event parameter,
- checkFilters** - executed defined filters for a task,
- manipulate** - executed the given manipulation on the given string value.

Methods

checkFilters

public boolean **checkFilters** ([List](#)<[Filter](#)> *filters*, [TaskContext](#) *taskContext*)

7.90.12 TaskInitializer

class **TaskInitializer**

The `TaskInitializer` class prepares an action in the task definition to execution.

- evalConfigSteps** - executes all config steps (load data sources, check filters) defined in the task,

See also: [TaskTriggerHandler](#), [TaskActionExecutor](#)

Constructors

TaskInitializer

TaskInitializer ([TaskContext](#) *taskContext*)

Methods

evalConfigSteps

public boolean **evalConfigSteps** ([Map](#)<[String](#), [DataProvider](#)> *dataProviders*)

7.90.13 TaskService

public interface **TaskService**

Methods

deleteTask

void **deleteTask** (Long *taskId*)

exportTask

String **exportTask** (Long *taskId*)

findTasksDependentOnModule

List<Task> **findTasksDependentOnModule** (String *moduleName*)

findTasksForTrigger

List<Task> **findTasksForTrigger** (TriggerEvent *trigger*)

findTasksForTriggerSubject

List<Task> **findTasksForTriggerSubject** (String *subject*)

findTrigger

TriggerEvent **findTrigger** (String *subject*)

getActionEventFor

ActionEvent **getActionEventFor** (TaskActionInformation *taskActionInformation*)

getAllTasks

List<Task> **getAllTasks** ()

getTask

Task **getTask** (Long *taskId*)

importTask

void **importTask** (String *json*)

save

void **save** ([Task](#) *task*)

7.90.14 TaskTriggerHandler

public class **TaskTriggerHandler** implements [TriggerHandler](#)

The `TaskTriggerHandler` receives events and executes tasks for which the trigger event subject is the same as the received event subject.

Constructors

TaskTriggerHandler

public **TaskTriggerHandler** ([TaskService](#) *taskService*, [TaskActivityService](#) *activityService*, [EventListenerRegistryService](#) *registryService*, [EventRelay](#) *eventRelay*, [SettingsFacade](#) *settings*)

Methods

addDataProvider

public void **addDataProvider** ([DataProvider](#) *provider*)

handle

public void **handle** ([MotechEvent](#) *event*)

registerHandlerFor

public final void **registerHandlerFor** ([String](#) *subject*)

removeDataProvider

public void **removeDataProvider** ([String](#) *taskDataProviderId*)

setBundleContext

public void **setBundleContext** ([BundleContext](#) *bundleContext*)

setDataProviders

void **setDataProviders** ([Map](#)<[String](#), [DataProvider](#)> *dataProviders*)

7.90.15 TriggerHandler

public interface **TriggerHandler**

Standard interface for tasks which executes tasks.

Methods

handle

void **handle** (*MotechEvent event*)

registerHandlerFor

void **registerHandlerFor** (*String subject*)

7.91 org.motechproject.tasks.util

7.91.1 BundleContextUtil

public final class **BundleContextUtil**

Methods

getSymbolicNames

public static *List<String>* **getSymbolicNames** (*BundleContext context*)

7.91.2 ManagementDataProvider

public class **ManagementDataProvider** implements *OsgiServiceLifecycleListener*

Constructors

ManagementDataProvider

public **ManagementDataProvider** (*TaskDataProviderService taskDataProviderService*)

ManagementDataProvider

public **ManagementDataProvider** (*TaskTriggerHandler handler*, *TaskDataProviderService taskDataProviderService*)

Methods

bind

public void **bind** (*Object service*, *Map serviceProperties*)

unbind

public void **unbind** (*Object service*, *Map serviceProperties*)

Contribute

Thank you for your interest in contributing to MOTECH. There are many different ways to get involved - regardless of your area of expertise and time commitment, there is likely a useful way for you to contribute. Please peruse the sections below for some instructions on how to get started contributing in specific areas. If there is another way you'd like to help that isn't listed, just [let us know](#) what your interests are and we can help find a project for you.

8.1 Development

Want to help us develop MOTECH? The step by step guide below will help you get started. The instructions may be incomplete, or may contain some errors, so please just let us know if something is missing or incorrect. Our mailing list is populated with helpful people who will help you get going - and will get this documentation up to date with any issues you encounter.

Here is how to get started:

8.1.1 Mailing Lists and Accounts

1. Sign up for the MOTECH Developer [mailing list](#) - this is the best place to get help with any questions about MOTECH development.
2. Create an account on [Gerrit](#), our code review system.
3. Get a Jira account if you'll need to report/track issues - there is no self-serve way to request a Jira account yet, so please just email the MOTECH Developer list and we'll get you set up.

8.1.2 Dev Environment & Tools

1. Configure your *dev machine*.
2. Configure your Git client to *submit changes via Gerrit*.
3. Familiarize yourself with our [CI](#).

8.1.3 Finding Something to Work On

1. Some MOTECH devs have found that reviewing code written by others before they jump in and develop their own is a good way to get their feet wet. If that sounds like your style, feel free to jump into any code review on Gerrit and add your comments.

2. If you'd rather dive right in, we'd recommend that you find a "community" issue from our issue tracker - these are bugs and user stories that we think are good introductory items for MOTECH newbies. See a list of all current "community" issues [here](#).
3. If you're already building your own system on top of MOTECH and you'd like us to incorporate your changes for an issue you've found and fixed, please first check whether the issue already exists in our issue tracker. If you are not sure, please email the mailing list and we'll help you determine whether the issue is already known. Please track your work with a new issue so that we can evaluate it for inclusion in the Platform.

8.1.4 Developing and Submitting Your Code

1. If your fix will be nontrivial, please leverage the mailing list for feedback on your design before you get too far - we are a friendly bunch and can help ensure you are headed in the right direction.
2. When you're ready to push your changes, please squash your commits to keep the change history concise, and write a commit message that conforms to our [guidelines](#).
3. Submit your code using **git push origin** - if you configured your environment correctly, this sends your changes to Gerrit, our code review system.
4. Please incorporate code review feedback and update your patch as needed - once your change has passed code review, one of the project maintainers will merge your change to the repo.
5. Resolve the relevant issue(s) in Jira.

8.2 Documentation

There will be text here.

8.3 Translation

Bonjour!

We are using [Transifex](#) to manage MOTECH translations, which makes it easy for non-geeks to help. If you speak multiple languages and would like to help us make MOTECH multilingual, please start by checking out our translation page to determine whether your language(s) are on our list. Do we have a translation in progress for your language? Great! We'd love your help translating additional strings. Do we not have a translation started for your language? Also great! We'd love your help getting one started.

Either way, please [sign up for Transifex](#) (free), and then [contact us](#). Let us know your Transifex user ID and which language(s) you'd like to work on, and we'll help you get started.

Indices and tables

- `genindex`
- `modindex`
- `search`

A

- abandonChanges(Long) (Java method), 256, 268
- AbstractDataProvider (Java class), 54
- AbstractDBConfig (Java class), 83
- AbstractDBConfig(String, String, String, String) (Java constructor), 83
- AbstractObjectValueGenerator (Java class), 229
- AbstractValidator (Java interface), 423
- accept(File) (Java method), 91
- accept(TaskActionInformation) (Java method), 435
- accessdenied(HttpServletRequest) (Java method), 411
- ACTION_PARAMETERS_FIELD (Java field), 481
- ActionEvent (Java class), 435
- ActionEvent() (Java constructor), 435
- ActionEvent(ActionEventRequest) (Java constructor), 435
- ActionEvent(String, String, String, SortedSet) (Java constructor), 435
- ActionEvent(String, String, String, String, SortedSet) (Java constructor), 435
- ActionEvent(String, String, String, String, String, SortedSet) (Java constructor), 435
- ActionEventRequest (Java class), 428
- ActionEventRequest(String, String, String, String, String) (Java constructor), 429
- ActionEventRequest(String, String, String, String, String, SortedSet) (Java constructor), 429
- ActionEventRequestDeserializer (Java class), 480
- ActionNotFoundException (Java class), 479
- ActionNotFoundException(String) (Java constructor), 479
- ActionParameter (Java class), 437
- ActionParameter() (Java constructor), 437
- ActionParameter(ActionParameterRequest) (Java constructor), 437
- ActionParameter(String, String) (Java constructor), 437
- ActionParameter(String, String, boolean) (Java constructor), 437
- ActionParameter(String, String, Integer) (Java constructor), 437
- ActionParameter(String, String, ParameterType) (Java constructor), 437
- ActionParameter(String, String, ParameterType, boolean) (Java constructor), 437
- ActionParameter(String, String, ParameterType, Integer) (Java constructor), 437
- ActionParameter(String, String, ParameterType, Integer, boolean) (Java constructor), 438
- ActionParameterRequest (Java class), 430
- ActionParameterRequest(String, String) (Java constructor), 430
- ActionParameterRequest(String, String, Integer) (Java constructor), 430
- ActionParameterRequest(String, String, Integer, String) (Java constructor), 430
- ActionParameterRequest(String, String, Integer, String, boolean) (Java constructor), 430
- ActionType (Java enum), 48
- ACTIVATE_USER_PERMISSION (Java field), 356
- activateUser(String) (Java method), 386
- Activator (Java class), 300
- ACTIVE (Java field), 393
- activeTasks() (Java method), 484
- ACTIVITY_ACTIVE (Java field), 332
- ACTIVITY_FINISHED (Java field), 332
- ACTIVITY_NOTSTARTED (Java field), 332
- add(AbstractValidator) (Java method), 426
- add(MotechPermission) (Java method), 377
- add(MotechRole) (Java method), 378
- add(MotechUser) (Java method), 380
- add>PasswordRecovery (Java method), 381
- add(String) (Java method), 90
- add(T) (Java method), 63
- add(TaskConfigStep) (Java method), 467
- ADD_NEW_INDEX (Java field), 180
- ADD_USER_PERMISSION (Java field), 356
- addAction(String) (Java method), 205
- addAction(TaskActionInformation) (Java method), 459, 466
- addActionTaskEvent(ActionEvent) (Java method), 439
- addAll(SortedSet) (Java method), 467

`addAngularModule(String)` (Java method), 306
`addBaseMetadata(JDOMetadata, ClassData)` (Java method), 129
`addBundle(Bundle)` (Java method), 303
`addClass(ClassData)` (Java method), 211
`addConfigLocation(String)` (Java method), 92
`addCriterion(ValidationCriterionDto)` (Java method), 193
`addData(Map)` (Java method), 63
`addData(String, Object)` (Java method), 63
`addDatabaseSuggestion(String)` (Java method), 419
`addDataProvider(DataProvider)` (Java method), 491
`addDataSource(DataSource)` (Java method), 466
`addDataSourceObject(String, Object, boolean)` (Java method), 488
`addDisplayedField(Number)` (Java method), 178
`addDisplayedFields(EntityDto, Map)` (Java method), 256, 269
`addEmptyMetadata()` (Java method), 189
`addEntityMetadata(JDOMetadata, Entity)` (Java method), 130
`addError(Task, TaskHandlerException)` (Java method), 486
`addField(Field)` (Java method), 142
`addField(Integer)` (Java method), 195
`addField(Long)` (Java method), 194
`addField(Number)` (Java method), 200, 205
`addFields(EntityDto, Collection)` (Java method), 256, 269
`addFilter(Filter)` (Java method), 448
`addFilterableField(Number)` (Java method), 178
`addFilterableFields(EntityDto, Collection)` (Java method), 256, 269
`addFilterSet(FilterSet)` (Java method), 466
`addHelperClassMetadata(JDOMetadata, ClassData, Entity)` (Java method), 130
`addI18N(String, String)` (Java method), 306
`addingService(ServiceReference)` (Java method), 302, 304, 317
`ADDITIONAL_DATA_PREFIX` (Java field), 448
`addLookup(Lookup)` (Java method), 143
`addLookup(Number)` (Java method), 201
`addLookups(Long, Collection)` (Java method), 256, 269
`addMetadata(FieldMetadata)` (Java method), 152
`addMetadata(MetadataDto)` (Java method), 189
`addNewIndex(String)` (Java method), 177
`addOpenIdUser(MotechUser)` (Java method), 380
`addOrReplace(T, String, String)` (Java method), 65
`addOrUpdate(Channel)` (Java method), 483
`addOrUpdate(ConfigSettings)` (Java method), 245
`addOrUpdate(File)` (Java method), 96
`addOrUpdate(MotechSecurityConfiguration)` (Java method), 379
`addOrUpdateModuleRecord(ModulePropertiesRecord)` (Java method), 96
`addOrUpdateModuleRecords(List)` (Java method), 96
`addOrUpdateProperties(String, String, String, String, Properties, Properties)` (Java method), 96
`addParameter(ActionParameter, boolean)` (Java method), 436
`addParameter(ActionParameterRequest, boolean)` (Java method), 429
`addPermission(PermissionDto)` (Java method), 384
`addProcessedRelation(String)` (Java method), 248
`addQueueSuggestion(String)` (Java method), 419
`addSchedulerSuggestion(String)` (Java method), 419
`addSetting(FieldSetting)` (Java method), 152
`addSubMenu(String, String)` (Java method), 306
`addSubMenu(String, String, String)` (Java method), 306
`addSuccess(Task)` (Java method), 487
`addTrackerFor(Bundle)` (Java method), 304
`addTrackerFor(Bundle, ApplicationContext)` (Java method), 317
`addValidation(FieldValidation)` (Java method), 152
`addValidationErrors(Set)` (Java method), 459
`addWarning(Task)` (Java method), 487
`addWarning(Task, String, String)` (Java method), 487
`addWarning(Task, String, String, Exception)` (Java method), 487
`ADMIN_CONFIRM_PASSWORD` (Java field), 416
`ADMIN_LOGIN` (Java field), 416
`ADMIN_PASSWORD` (Java field), 416
`AdminSettings` (Java class), 43
`AdminSettings(List, boolean)` (Java constructor), 43
`ADVANCED` (Java field), 180
`AdvancedSettingsDto` (Java class), 176
`advancedSettingsDto()` (Java method), 143
`AFTER` (Java field), 455
`AFTER_NOW` (Java field), 455
`afterPropertiesSet()` (Java method), 235, 396
`ALL` (Java field), 222, 453
`AllConfigSettings` (Java class), 244
`AllConfigSettings()` (Java constructor), 245
`AllEntities` (Java class), 245
`AllEntities()` (Java constructor), 245
`AllEntityAudits` (Java class), 246
`AllEntityAudits()` (Java constructor), 246
`AllEntityDrafts` (Java class), 246
`AllEntityDrafts()` (Java constructor), 246
`AllMotechPermissions` (Java class), 377
`AllMotechRoles` (Java class), 378
`AllMotechSecurityRules` (Java class), 379
`AllMotechUsers` (Java class), 379
`ALLOW_MULTIPLE_SELECTIONS` (Java field), 287
`ALLOW_USER_SUPPLIED` (Java field), 287
`AllPasswordRecoveries` (Java class), 381
`allRecoveries()` (Java method), 381
`AllTypes` (Java class), 248
`AllTypes()` (Java constructor), 248
`AllTypeSettings` (Java class), 247

- AllTypeSettings() (Java constructor), 247
 - AllTypeValidations (Java class), 247
 - AllTypeValidations() (Java constructor), 247
 - AlreadyRegisteredException (Java class), 410
 - AlreadyRegisteredException(String) (Java constructor), 410
 - AMQ_BROKER_URL (Java field), 80
 - AMQ_CONCURRENT_CONSUMERS (Java field), 80
 - AMQ_MAX_CONCURRENT_CONSUMERS (Java field), 80
 - AMQ_MAX_REDELIVERIES (Java field), 81
 - AMQ_QUEUE_EVENTS (Java field), 81
 - AMQ_QUEUE_SCHEDULER (Java field), 81
 - AMQ_REDELIVERY_DELAY_IN_MILLIS (Java field), 81
 - AnnotationFields (Java class), 278
 - ANY (Java field), 355
 - ANY_PATTERN (Java field), 358
 - ApplicationContextServiceReferenceUtils (Java class), 55
 - ApplicationContextTracker (Java class), 300
 - ApplicationContextTracker(BundleContext) (Java constructor), 301
 - ApplicationEnvironment (Java class), 320
 - areConfigurationSettingsRegistered() (Java method), 397
 - ASC (Java field), 295
 - ascOrder(String) (Java method), 241
 - asDeclareParameter(int) (Java method), 238
 - asFilter(int) (Java method), 238
 - asMatchesPattern(String) (Java method), 242
 - asPeriod() (Java method), 78
 - asProperties() (Java method), 132, 397, 400, 403
 - asProperties(Map) (Java method), 58
 - assertArgumentNotEmpty(String, String) (Java method), 59
 - assertArgumentNotNull(String, Object) (Java method), 59, 348
 - authoritiesFor(MotechUser) (Java method), 383
 - AuthoritiesService (Java interface), 383
- ## B
- BASE (Java field), 284
 - BaseDao (Java interface), 63
 - BasePersistenceService (Java class), 267
 - BASIC (Java field), 357
 - BEFORE (Java field), 455
 - BEFORE_NOW (Java field), 455
 - bind(Object, Map) (Java method), 493
 - BLANK (Java field), 473
 - BLOB (Java field), 206
 - BLUEPRINT_TEMPLATE (Java field), 260
 - BLUEPRINT_XML (Java field), 260
 - BlueprintActivator (Java class), 301
 - BlueprintApplicationContextTracker (Java class), 301
 - BlueprintApplicationContextTracker(BundleContext) (Java constructor), 302
 - BOOLEAN (Java field), 206, 457
 - BootstrapConfig (Java class), 84
 - BootstrapConfig(DBConfig, SQLDBConfig, String, ConfigSource) (Java constructor), 85
 - breakString(String) (Java method), 298
 - breakString(String, String[], String[], String[], String) (Java method), 298
 - breakStringForList(String) (Java method), 298
 - BrowsingSettings (Java class), 137
 - BrowsingSettings(Entity) (Java constructor), 137
 - BrowsingSettingsDto (Java class), 178
 - build() (Java method), 66, 327, 328, 466
 - build(ComboBoxHolder) (Java method), 130
 - build(Entity) (Java method), 127
 - buildDDE(Entity, Bundle) (Java method), 128
 - buildFrom(File) (Java method), 93
 - buildHistory(Entity) (Java method), 128
 - buildHistoryInfrastructure(String) (Java method), 129
 - buildInfrastructure(Entity) (Java method), 129
 - buildMenu(String) (Java method), 423
 - buildSecurityChain(MotechURLSecurityRule, HTTPMethod) (Java method), 354
 - buildTrash(Entity) (Java method), 128
 - bulkAddOrUpdate(List) (Java method), 65
 - bulkDelete(List) (Java method), 65
 - BUNDLE_ID (Java field), 81
 - BUNDLE_IMPORTS (Java field), 260
 - BUNDLE_MANIFESTVERSION (Java field), 282
 - BUNDLE_NAME (Java field), 391
 - BUNDLE_NAME_SUFFIX (Java field), 282
 - BUNDLE_SECTION (Java field), 81
 - BUNDLE_SETTINGS_CHANGED_EVENT_SUBJECT (Java field), 81
 - BUNDLE_SYMBOLIC_NAME (Java field), 81, 394
 - BUNDLE_VERSION (Java field), 394
 - BundleContextUtil (Java class), 492
 - BundleContextWrapper (Java class), 302
 - BundleContextWrapper() (Java constructor), 302
 - BundleContextWrapper(BundleContext) (Java constructor), 302
 - BundledJspView (Java class), 303
 - BundleHeaders (Java class), 325
 - BundleHeaders(Bundle) (Java constructor), 325
 - BundleHeaders(BundleContext) (Java constructor), 325
 - BundleIcon (Java class), 390
 - BundleIcon(byte[], String) (Java constructor), 390
 - BundleInformation (Java class), 391
 - BundleInformation(Bundle) (Java constructor), 391
 - BundleLoader (Java interface), 393
 - BundleLoadingException (Java class), 394
 - BundleLoadingException(String) (Java constructor), 394

`BundleLoadingException(String, Throwable)` (Java constructor), 394
`BundleLoadingException(Throwable)` (Java constructor), 394
`BundleName` (Java class), 321
`BundleName(String)` (Java constructor), 321
`BundleNames` (Java class), 280
`BundleRegister` (Java class), 303
`BusinessIdNotUniqueException` (Java class), 64
`BusinessIdNotUniqueException(String, String)` (Java constructor), 64
`byTask(Long)` (Java method), 482

C

`callHandler(MotechEvent)` (Java method), 121–123
`canHaveQueue(String)` (Java method), 61
`canLaunchBundles()` (Java method), 409
`CAPITALIZE` (Java field), 454
`Cascade` (Java annotation), 125
`cast(Class, Enumeration)` (Java method), 56
`CastUtils` (Java class), 56
`changeLogLevel(String, String)` (Java method), 323
`changePassword(String, String, String)` (Java method), 386
`changeRootLogLevel(String)` (Java method), 323
`Channel` (Java class), 439
`Channel()` (Java constructor), 439
`Channel(ChannelRequest)` (Java constructor), 439
`Channel(String, String, String)` (Java constructor), 439
`Channel(String, String, String, String, List, List)` (Java constructor), 439
`ChannelRegisterEvent` (Java class), 441
`ChannelRegisterEvent(MotechEvent)` (Java constructor), 441
`ChannelRegisterEvent(String)` (Java constructor), 441
`ChannelRequest` (Java class), 432
`ChannelRequest(String, String, String, String, List, List)` (Java constructor), 432
`ChannelsDataService` (Java interface), 482
`ChannelService` (Java interface), 482
`checkFilters(List, TaskContext)` (Java method), 489
`checkListContainLogger(List, String)` (Java method), 305
`checkLogXmlConfiguration(Document)` (Java method), 305
`ClassData` (Java class), 138
`ClassData(Entity, byte[])` (Java constructor), 138
`ClassData(Entity, byte[], boolean)` (Java constructor), 138
`ClassData(String, byte[])` (Java constructor), 138
`ClassData(String, byte[], boolean)` (Java constructor), 138
`ClassData(String, String, String, byte[])` (Java constructor), 138
`ClassData(String, String, String, byte[], boolean)` (Java constructor), 138

`ClassData(String, String, String, byte[], boolean, EntityType)` (Java constructor), 138
`ClassData(String, String, String, byte[], EntityType)` (Java constructor), 138
`classForName(String)` (Java method), 231
`classForName(String, boolean)` (Java method), 232
`classForName(String, ClassLoader)` (Java method), 231, 233
`classForName(String, ClassLoader, boolean)` (Java method), 231
`ClassName` (Java class), 277
`cleanupExpiredRecoveries()` (Java method), 388
`clear()` (Java method), 466
`clearEnhancedData()` (Java method), 227
`clearListenersForBean(String)` (Java method), 120
`CollectionProperty` (Java class), 236
`CollectionProperty(String, Object)` (Java constructor), 236
`COMBOBOX_VALUES` (Java field), 287
`ComboboxHolder` (Java class), 139
`ComboboxHolder(Entity, Field)` (Java constructor), 139
`ComboboxHolder(EntityDto, FieldDto)` (Java constructor), 140
`ComboboxHolder(Field)` (Java constructor), 139
`ComboboxHolder(List, List, String)` (Java constructor), 140
`commence(HttpServletRequest, HttpServletResponse, AuthenticationException)` (Java method), 352, 353
`commitChanges(Long)` (Java method), 256, 269
`commitChanges(Long, String)` (Java method), 256, 269
`compare(EmailRecord, EmailRecord)` (Java method), 111
`compareTo(ActionParameter)` (Java method), 438
`compareTo(ActionParameterRequest)` (Java method), 431
`compareTo(TaskActivity)` (Java method), 464
`compareTo(TaskConfigStep)` (Java method), 469
`compareTo(Time)` (Java method), 71
`Config` (Java class), 281
`CONFIG_LOCATION_PROPERTY_KEY` (Java field), 90
`CONFIG_MODULE_DIR_PREFIX` (Java field), 81
`CONFIG_SOURCE` (Java field), 84
`ConfigFileFilter` (Java class), 91
`ConfigFileMonitor` (Java class), 95
`ConfigLoader` (Java class), 406
`ConfigLocation` (Java class), 86
`ConfigLocation(String)` (Java constructor), 86
`ConfigLocationFileStore` (Java class), 90
`ConfigLocationFileStore(PropertiesConfiguration)` (Java constructor), 90
`ConfigSettings` (Java class), 141
`ConfigSettings()` (Java constructor), 141
`ConfigSettings(DeleteMode, boolean, int, TimeUnit)` (Java constructor), 141

- ConfigSource (Java class), 88
- ConfigurationConstants (Java class), 80
- ConfigurationService (Java interface), 96
- Constants (Java class), 278, 410
- constructEntities(boolean) (Java method), 131
- constructSymbolicName() (Java method), 397
- CONTAINS (Java field), 455
- contains(String) (Java method), 63, 245
- containsAction(TaskActionInformation) (Java method), 439
- containsBean(String) (Java method), 311
- containsBeanDefinition(String) (Java method), 311
- containsDeclaredField(CtClass, String) (Java method), 224
- containsDeclaredMethod(CtClass, String) (Java method), 224
- containsDisplayedField(Long) (Java method), 179
- containsField(CtClass, String) (Java method), 225
- containsField(String) (Java method), 471
- containsFieldId(Number) (Java method), 201
- containsFilterableField(Number) (Java method), 179
- containsLocalBean(String) (Java method), 311
- containsLookupId(Number) (Java method), 201
- containsMethod(CtClass, String) (Java method), 225
- containsOnlyNullValues(Collection) (Java method), 238
- containsParameter(String) (Java method), 436, 474, 478
- containsProviderObject(String) (Java method), 470
- containsProviderObjectField(String, String) (Java method), 470
- containsProviderObjectLookup(String, String) (Java method), 470
- containsTrigger(TaskTriggerInformation) (Java method), 439
- CONTEXT_SERVICE_NAME (Java field), 302
- contextInvalidOrProcessed(ServiceReference, ApplicationContext) (Java method), 301
- convertToCSV(List) (Java method), 56
- copy() (Java method), 152, 157, 159, 160, 168, 169, 203
- copy(List) (Java method), 162
- copy(String, Map) (Java method), 118
- CORE_SETTINGS_CACHE_NAME (Java field), 91
- CoreConfigurationService (Java interface), 91
- COUCHDB_PASSWORD (Java field), 84
- COUCHDB_URL (Java field), 84
- COUCHDB_USERNAME (Java field), 85
- CouchDbLuceneQuery (Java class), 66
- CouchDbManager (Java interface), 68
- count() (Java method), 252, 262
- count(InstanceSecurityRestriction) (Java method), 249
- count(List) (Java method), 253, 264
- count(List, InstanceSecurityRestriction) (Java method), 249
- count(String[], Object[], InstanceSecurityRestriction) (Java method), 249
- countEmailRecords(EmailRecordSearchCriteria) (Java method), 114
- countFind(String, String, String, String, Range, Set) (Java method), 115
- countForFilter(Filter) (Java method), 249, 253, 262
- countForFilter(Filter, InstanceSecurityRestriction) (Java method), 249
- countHistoryRecords(Object) (Java method), 259, 272
- countTrashRecords(String) (Java method), 264, 275
- create(Class, Object, EntityType) (Java method), 267, 272
- create(Class, Object, EntityType, ObjectReference) (Java method), 267
- create(Entity, String) (Java method), 246
- create(EntityDto) (Java method), 245
- create(Field, Object) (Java method), 239
- create(String, Object) (Java method), 239
- create(String, Object, String) (Java method), 239
- create(T) (Java method), 249, 253, 262
- createApplicationContext(BundleContext) (Java method), 53
- createAudit(Entity, String) (Java method), 246
- createEntity(EntityDto) (Java method), 256, 269
- createEnumInitializer(String, String) (Java method), 223
- createField(CtClass, CtClass, String, String) (Java method), 223
- createGetter(String, CtClass, CtField) (Java method), 223
- createInitializer(String, String) (Java method), 223
- createListInitializer(String, Object) (Java method), 224
- createLocaleInitializer(String) (Java method), 224
- createLoggerProperties(List) (Java method), 305
- createParameters(TaskActionInformation, ActionEvent) (Java method), 486
- createRecovery(String, String, String, DateTime, Locale) (Java method), 381
- createRole(RoleDto) (Java method), 385
- createSetter(String, CtField) (Java method), 224
- createSimpleInitializer(String, Object) (Java method), 224
- createSimpleInitializer(String, String) (Java method), 224
- createZipWithConfigFiles(String, String) (Java method), 97
- CREATION_DATE_DISPLAY_FIELD_NAME (Java field), 288
- CREATION_DATE_FIELD_NAME (Java field), 288
- CreationDateValueGenerator (Java class), 230
- CREATOR (Java field), 297
- CREATOR_DISPLAY_FIELD_NAME (Java field), 288
- CREATOR_FIELD_NAME (Java field), 288
- CreatorValueGenerator (Java class), 230
- CRITICAL (Java field), 49
- critical(String, String) (Java method), 50
- critical(String, String, DateTime) (Java method), 50
- CronJobExpressionBuilder (Java class), 327

- CronJobExpressionBuilder(Time, Integer, Integer) (Java constructor), 327
- CronJobId (Java class), 328
- CronJobId(MotechEvent) (Java constructor), 328
- CronJobId(String, String) (Java constructor), 328
- CronJobSimpleExpressionBuilder (Java class), 327
- CronJobSimpleExpressionBuilder(Time) (Java constructor), 327
- CronSchedulableJob (Java class), 329
- CronSchedulableJob(MotechEvent, String) (Java constructor), 329
- CronSchedulableJob(MotechEvent, String, Date, Date) (Java constructor), 329
- CronSchedulableJob(MotechEvent, String, Date, Date, boolean) (Java constructor), 329
- CsvConverter (Java class), 56
- current() (Java method), 61
- currentVersion(Class) (Java method), 289
- CustomOperatorProperty (Java class), 237
- CustomOperatorProperty(String, T, String) (Java constructor), 237
- D**
- DashboardController (Java class), 410
- DATA_ACCESS (Java field), 286
- DATANUCLEUS (Java field), 288
- DATANUCLEUS_FILE (Java field), 281
- DATANUCLEUS_PROPERTIES (Java field), 260
- DataProvider (Java interface), 56
- DataProviderDataService (Java interface), 483
- DataSource (Java class), 442
- DataSource() (Java constructor), 442
- DataSource(Long, Long, String, String, List, boolean) (Java constructor), 442
- DataSource(String, Long, Long, String, String, List, boolean) (Java constructor), 442
- DataSourceObject (Java class), 483
- DataSourceObject(String, Object, boolean) (Java constructor), 484
- DATE (Java field), 207, 453, 457
- DATE_TIME_FORMAT (Java field), 66
- DATETIME (Java field), 207, 454
- DateTimeSource (Java interface), 77
- DateTimeSourceUtil (Java class), 72
- DateTimeValueGenerator (Java class), 231
- DateUtil (Java class), 73
- Day (Java field), 79
- DayOfWeek (Java enum), 69
- DayOfWeekSchedulableJob (Java class), 330
- DayOfWeekSchedulableJob(MotechEvent, LocalDate, LocalDate, List, Time) (Java constructor), 330
- DayOfWeekSchedulableJob(MotechEvent, LocalDate, LocalDate, List, Time, boolean) (Java constructor), 330
- DAYS (Java field), 136
- daysPast(LocalDate, DayOfWeek) (Java method), 73
- daysStarting(DayOfWeek, int) (Java method), 73
- daysToCalendarWeekEnd(LocalDate, int) (Java method), 73
- DB_ERROR (Java field), 408
- DBConfig (Java class), 89
- DBConfig(String, String, String) (Java constructor), 89
- DbConnectionException (Java class), 69
- DbConnectionException(String) (Java constructor), 69
- DbConnectionException(String, Throwable) (Java constructor), 69
- DEBUG (Java field), 49
- debug(String, String) (Java method), 50
- debug(String, String, DateTime) (Java method), 50
- DEFAULT_ACTIVEMQ_URL (Java field), 423
- DEFAULT_DELETE_MODE (Java field), 133
- DEFAULT_EMPTY_TRASH (Java field), 133
- DEFAULT_TENANT_ID (Java field), 85
- DEFAULT_TIME_UNIT (Java field), 133
- DEFAULT_TIME_VALUE (Java field), 133
- DEFAULT_WAIT_TIME (Java field), 274
- DefaultDateTimeSource (Java class), 77
- DefaultDateTimeSource() (Java constructor), 77
- DefaultMotechDataService (Java class), 252
- defineClass(String, byte[]) (Java method), 292
- DELETE (Java field), 132, 278, 355
- delete(EmailRecord) (Java method), 114
- delete(Long) (Java method), 245
- delete(MotechPermission) (Java method), 377
- delete(String) (Java method), 97
- delete(String, Object) (Java method), 253, 262
- delete(String, Object, InstanceSecurityRestriction) (Java method), 250
- delete(String[], Object[], InstanceSecurityRestriction) (Java method), 250
- delete(T) (Java method), 249, 253, 262
- DELETE_USER_PERMISSION (Java field), 356
- deleteActivitiesForTask(Long) (Java method), 487
- deleteAll() (Java method), 253, 263
- deleteAll(Entity) (Java method), 246
- deleteAll(String, Object) (Java method), 250
- deleteAll(String, Object, InstanceSecurityRestriction) (Java method), 250
- deleteAll(String[], Object[], InstanceSecurityRestriction) (Java method), 250
- deleteByBundle(String) (Java method), 97
- deleteEntity(Long) (Java method), 256, 269
- DeleteMode (Java enum), 131
- deletePermission(String) (Java method), 384
- deleteRole(RoleDto) (Java method), 385
- deleteTask(Long) (Java method), 490
- deleteUser(UserDto) (Java method), 386
- DeliveryStatus (Java enum), 108

DEP_WAIT_TIME_ENV (Java field), 52
 DEP_WAIT_TIME_KEY (Java field), 52
 DESC (Java field), 295
 descOrder(String) (Java method), 241
 DESCRIPTION_FIELD (Java field), 481
 deserialize(JsonElement, Type, JsonDeserializationContext) (Java method), 481
 deserialize(JsonParser, DeserializationContext) (Java method), 135, 481, 482
 Deserializer (Java class), 135
 DEVELOPMENT (Java field), 320
 Direction (Java enum), 295
 DISPLAY_NAME (Java field), 180, 278
 DISPLAY_NAME_FIELD (Java field), 481
 doBegin(Object, TransactionDefinition) (Java method), 234
 doCleanupAfterCompletion(Object) (Java method), 234
 doCreateConnection() (Java method), 124
 doInTransaction(TransactionCallback) (Java method), 253, 263
 DOUBLE (Java field), 192, 207, 458
 doWhenClassNotFound(String) (Java method), 227, 291
 DraftData (Java class), 180
 DraftResult (Java class), 182
 DraftResult(boolean, boolean) (Java constructor), 182
 duration() (Java method), 60

E

EDIT_USER_PERMISSION (Java field), 356
 ElementOrder (Java class), 421
 EMAIL (Java field), 49
 EmailAuditService (Java interface), 113
 EmailExistsException (Java class), 368
 EmailExistsException(String) (Java constructor), 368
 EmailRecord (Java class), 108
 EmailRecord() (Java constructor), 108
 EmailRecord(String, String, String, String, DateTime, DeliveryStatus) (Java constructor), 109
 EmailRecordComparator (Java class), 111
 EmailRecordComparator(Boolean, String) (Java constructor), 111
 EmailRecords (Java class), 112
 EmailRecords() (Java constructor), 112
 EmailRecords(Integer, Integer, Integer, List) (Java constructor), 112
 EmailRecordSearchCriteria (Java class), 103
 EmailRecordService (Java interface), 115
 EmailSenderService (Java interface), 116
 EMPTY_COLLECTION (Java field), 474
 EMPTY_TRASH_JOB (Java field), 281
 emptyTrash() (Java method), 265, 275
 EmptyTrashException (Java class), 212
 EmptyTrashException(Throwable) (Java constructor), 212
 encodePassword(String) (Java method), 352
 endOfDay(Date) (Java method), 73
 ENDS_WITH (Java field), 283
 ENDSWITH (Java field), 455
 Entity (Java annotation), 125
 Entity (Java class), 142
 ENTITY (Java field), 284, 285, 288
 Entity() (Java constructor), 142
 Entity(String) (Java constructor), 142
 Entity(String, String, String, SecurityMode) (Java constructor), 142
 Entity(String, String, String, String, SecurityMode, Set) (Java constructor), 142
 ENTITY_LIST_FILE (Java field), 260
 EntityAlreadyExistException (Java class), 212
 EntityAlreadyExistException() (Java constructor), 212
 EntityAudit (Java class), 147
 EntityBuilder (Java interface), 127
 EntityChangedException (Java class), 212
 EntityChangedException() (Java constructor), 212
 EntityCreationException (Java class), 212
 EntityCreationException(String) (Java constructor), 213
 EntityCreationException(Throwable) (Java constructor), 213
 EntityDeletedException (Java class), 213
 EntityDeletedException() (Java constructor), 213
 EntityDraft (Java class), 148
 EntityDraft() (Java constructor), 148
 EntityDto (Java class), 183
 EntityDto() (Java constructor), 183
 EntityDto(Long, String) (Java constructor), 183
 EntityDto(Long, String, SecurityMode, Set) (Java constructor), 183
 EntityDto(Long, String, String, SecurityMode, Set) (Java constructor), 183
 EntityDto(Long, String, String, String, SecurityMode, Set) (Java constructor), 183
 EntityDto(Long, String, String, String, String, SecurityMode, Set) (Java constructor), 184
 EntityDto(Long, String, String, String, String, SecurityMode, Set, String) (Java constructor), 184
 EntityDto(Long, String, String, String, String, SecurityMode, Set, String, boolean) (Java constructor), 184
 EntityDto(String) (Java constructor), 183
 EntityDto(String, SecurityMode, Set) (Java constructor), 183
 EntityDto(String, String, String, String, SecurityMode, Set) (Java constructor), 183
 EntityInfo (Java class), 150
 EntityInfrastructureBuilder (Java interface), 129
 EntityInfrastructureException (Java class), 213
 EntityInfrastructureException(Throwable) (Java constructor), 213

- EntityMetadataBuilder (Java interface), 129
 - EntityNotFoundException (Java class), 213
 - EntityNotFoundException() (Java constructor), 214
 - EntityReadOnlyException (Java class), 214
 - EntityReadOnlyException() (Java constructor), 214
 - EntitySchemaMismatchException (Java class), 214
 - EntitySchemaMismatchException() (Java constructor), 214
 - EntityService (Java interface), 255
 - EntityServiceImpl (Java class), 268
 - EntityType (Java enum), 151
 - ENUM_CLASS_NAME (Java field), 283
 - EnumBuilder (Java interface), 130
 - ENVIRONMENT (Java field), 320
 - EQ (Java field), 283
 - EQ_IGNORE_CASE (Java field), 283
 - EQ_NUMBER (Java field), 455
 - EqualProperty (Java class), 237
 - EqualProperty(String, T) (Java constructor), 237
 - EQUALS (Java field), 455
 - equals(Object) (Java method), 60, 71, 78, 83, 86, 87, 93, 106, 109, 118, 133, 177, 179, 184, 187, 189, 191, 193, 195, 197, 199, 201, 203, 205, 208, 210, 306, 319, 321, 324, 329, 336, 339, 364, 370, 371, 391, 420, 425, 429, 431–434, 436, 438, 439, 442, 444–446, 448, 449, 452, 457, 459, 462, 464, 467, 469–471, 473, 474, 476, 478, 484
 - equalsSubject(String) (Java method), 474
 - ERROR (Java field), 49, 108, 465
 - error(String, String) (Java method), 50
 - error(String, String, DateTime) (Java method), 50
 - ERROR_REQUIRED (Java field), 425
 - errorsFromLastRun(Task) (Java method), 487
 - evalConfigSteps(Map) (Java method), 489
 - evaluate(Object) (Java method), 294
 - evaluateTemplateString(String) (Java method), 485
 - EVENT_RELAY_CLASS_NAME (Java field), 81
 - EVENT_TYPE_KEY_NAME (Java field), 116
 - EventInfo (Java class), 331
 - EventInfo() (Java constructor), 331
 - EventListener (Java interface), 119
 - EventListenerRegistryService (Java interface), 119
 - EventParameter (Java class), 444
 - EventParameter() (Java constructor), 444
 - EventParameter(EventParameterRequest) (Java constructor), 444
 - EventParameter(String, String) (Java constructor), 444
 - EventParameter(String, String, ParameterType) (Java constructor), 444
 - EventParameterRequest (Java class), 433
 - EventParameterRequest(String, String) (Java constructor), 433
 - EventParameterRequest(String, String, String) (Java constructor), 433
 - EventRelay (Java interface), 121
 - EVERYONE (Java field), 297
 - evictMotechCoreSettingsCache() (Java method), 92
 - evictMotechSettingsCache() (Java method), 97
 - execute(JobExecutionContext) (Java method), 274, 346
 - execute(Query, InstanceSecurityRestriction) (Java method), 240
 - execute(Query, Object, InstanceSecurityRestriction) (Java method), 240
 - execute(Task, TaskActionInformation, TaskContext) (Java method), 486
 - executeQuery(QueryExecution) (Java method), 253, 263
 - executeWithArray(Query, List) (Java method), 240
 - executeWithArray(Query, Object[], InstanceSecurityRestriction) (Java method), 240
 - executeWithFilter(Query, Filter, InstanceSecurityRestriction) (Java method), 240
 - EXIST (Java field), 456
 - exists(String, Object) (Java method), 250
 - exists(String[], Object[]) (Java method), 250
 - exportPlatformSettings() (Java method), 407
 - exportTask(Long) (Java method), 490
 - ExtensibleDataObject (Java class), 63
 - EXTRACTION_FAILED (Java field), 394
 - extractJarInformationFromPath() (Java method), 396
- ## F
- FALSE (Java field), 288
 - Field (Java annotation), 126
 - Field (Java class), 151
 - FIELD (Java field), 180
 - Field() (Java constructor), 151
 - Field(Entity, String, String, boolean, boolean, String, String, Set) (Java constructor), 151
 - Field(Entity, String, String, Set) (Java constructor), 151
 - Field(Entity, String, String, Type, boolean, boolean) (Java constructor), 151
 - FIELD_ID (Java field), 180
 - FIELD_NAME_START_IDX (Java field), 292
 - FieldBasicDto (Java class), 186
 - FieldBasicDto() (Java constructor), 187
 - FieldBasicDto(String, String) (Java constructor), 187
 - FieldBasicDto(String, String, boolean, Object, String) (Java constructor), 187
 - FieldDto (Java class), 188
 - FieldDto() (Java constructor), 188
 - FieldDto(Long, Long, TypeDto, FieldBasicDto, boolean, FieldValidationDto) (Java constructor), 189
 - FieldDto(Long, Long, TypeDto, FieldBasicDto, boolean, List, FieldValidationDto, List, List) (Java constructor), 188
 - FieldHelper (Java class), 289

- FieldHolder (Java class), 156
- FieldHolder(Field) (Java constructor), 156
- FieldHolder(FieldDto) (Java constructor), 156
- FieldHolder(List, List) (Java constructor), 156
- FieldInstanceDto (Java class), 191
- FieldInstanceDto() (Java constructor), 191
- FieldInstanceDto(Long, Long, FieldBasicDto) (Java constructor), 191
- FieldMetadata (Java class), 157
- FieldMetadata() (Java constructor), 157
- FieldMetadata(Field, String) (Java constructor), 157
- FieldMetadata(Field, String, String) (Java constructor), 157
- FieldMetadata(MetadataDto) (Java constructor), 157
- FieldNotFoundException (Java class), 214
- FieldNotFoundException() (Java constructor), 214
- FieldParameter (Java class), 445
- FieldParameter() (Java constructor), 445
- FieldParameter(String, String) (Java constructor), 445
- FieldParameter(String, String, ParameterType) (Java constructor), 445
- FieldSetting (Java class), 158
- FieldSetting() (Java constructor), 158
- FieldSetting(Field, TypeSetting) (Java constructor), 159
- FieldSetting(Field, TypeSetting, String) (Java constructor), 159
- FieldUsedInLookupException (Java class), 215
- FieldUsedInLookupException(String, String) (Java constructor), 215
- FieldValidation (Java class), 160
- FieldValidation() (Java constructor), 160
- FieldValidation(Field, TypeValidation) (Java constructor), 160
- FieldValidation(Field, TypeValidation, String, boolean) (Java constructor), 160
- FieldValidationDto (Java class), 192
- FieldValidationDto() (Java constructor), 193
- FieldValidationDto(ValidationCriterionDto) (Java constructor), 193
- FILE (Java field), 88
- FILE_CHANGED_EVENT_SUBJECT (Java field), 81
- FILE_CREATED_EVENT_SUBJECT (Java field), 82
- FILE_DELETED_EVENT_SUBJECT (Java field), 82
- FILE_PATH (Java field), 82
- FileAccessType (Java enum), 88
- fileChanged(FileChangeEvent) (Java method), 95
- fileCreated(FileChangeEvent) (Java method), 95
- fileDeleted(FileChangeEvent) (Java method), 95
- FileSystemAwareUIHttpContext (Java class), 321
- FileSystemAwareUIHttpContext(HttpContext, String) (Java constructor), 322
- Filter (Java class), 221, 446
- Filter() (Java constructor), 221, 446
- Filter(EventParameter, boolean, String, String) (Java constructor), 446
- filter(Filter) (Java method), 253, 263
- filter(Filter, QueryParams) (Java method), 250, 253, 263
- filter(Filter, QueryParams, InstanceSecurityRestriction) (Java method), 250
- Filter(String, FilterType) (Java constructor), 221
- Filter(String, String, ParameterType, boolean, String, String) (Java constructor), 446
- filterForQuery() (Java method), 221
- FilterSet (Java class), 447
- FilterSet() (Java constructor), 447
- FilterSet(List) (Java constructor), 448
- FilterType (Java enum), 222
- find(String, String, String, String, Range, Set, QueryParams) (Java method), 115
- findAllEmailRecords() (Java method), 114
- findAllSecurityRules() (Java method), 386
- findAnnotationOnBean(String, Class) (Java method), 311
- findBundleByName(BundleContext, String) (Java method), 326
- findBundleBySymbolicName(BundleContext, String) (Java method), 326
- findByBundle(String) (Java method), 102
- findByEmail(String) (Java method), 382
- findByExpirationDate(Range) (Java method), 383
- findById(Long) (Java method), 253, 263
- findById(long) (Java method), 114
- findByModule(String) (Java method), 103
- findByModuleAndFileName(String, String) (Java method), 103
- findByModuleName(String) (Java method), 482
- findByName(String) (Java method), 483
- findByOpenId(String) (Java method), 382
- findByOrigin(String) (Java method), 382
- findByPermissionName(String) (Java method), 378, 382
- findByRecipientAddress(String) (Java method), 116
- findByRole(String) (Java method), 380, 383
- findByRoleName(String) (Java method), 378, 382
- findByTimeout(Range) (Java method), 52
- findByUsername(String) (Java method), 380, 383
- findDeclaredField(CtClass, String) (Java method), 225
- findDeclaredMethod(CtClass, String) (Java method), 225
- findEmailRecords(EmailRecordSearchCriteria) (Java method), 114
- findEntityFieldByName(Long, String) (Java method), 256, 269
- findExistingConfigs() (Java method), 407
- findField(CtClass, String) (Java method), 225
- findFieldByName(Long, String) (Java method), 256, 269
- findFilename(String) (Java method), 397
- findForToken(String) (Java method), 381, 383
- findForUser(String) (Java method), 381, 383
- findMethod(CtClass, String) (Java method), 225

findService(Class) (Java method), 267
findTasksDependentOnModule(String) (Java method), 490
findTasksForTrigger(TriggerEvent) (Java method), 490
findTasksForTriggerSubject(String) (Java method), 490
findTrashById(Object, Object) (Java method), 265, 275
findTrashInstanceById(Object, Object) (Java method), 254, 263
findTrigger(String) (Java method), 490
findType(Class) (Java method), 266, 276
findUserByEmail(String) (Java method), 380
findUserByOpenId(String) (Java method), 380
findValidations(TypeDto, Class) (Java method), 266, 276
FIRST_RUN (Java field), 408
fixEnhancerIssuesInMetadata(JDOMetadata) (Java method), 130
ForgotController (Java class), 411
forgotPost(String) (Java method), 411
FORMAT (Java field), 454
format(Object) (Java method), 298
formatDateTime(DateTime) (Java method), 76
formatPeriod(Period) (Java method), 76
FRACTION (Java field), 278
Friday (Java field), 69
fromAdditionalData() (Java method), 449
fromHeaders(MessageHeaders, Message) (Java method), 125
fromTrigger() (Java method), 449

G

ge(Time) (Java method), 71
generate() (Java method), 261, 273
generate(ExecutionContext, Object, ExtensionMetadata[]) (Java method), 230
generateDataProvider() (Java method), 131
generateDeclareParameter(int) (Java method), 236, 239, 243, 244
generateFilter(int) (Java method), 236–239, 243, 244
generateHeader(Bundle) (Java method), 421
generateSchema() (Java method), 235
genericSignature(Class, Class) (Java method), 225
genericSignature(Class, String) (Java method), 225
genericSignature(String, String) (Java method), 225
GET (Java field), 355
get(Object) (Java method), 325
get(String) (Java method), 64
getAction(TaskActionInformation) (Java method), 440
getActionEventFor(TaskActionInformation) (Java method), 490
getActionParameters() (Java method), 429, 436
getActions() (Java method), 205, 460
getActionTaskEvents() (Java method), 432, 440
getActionType() (Java method), 44
getActiveMessages() (Java method), 50

getActivemqConfig() (Java method), 397
getActivemqProperties() (Java method), 400, 403
getActivity() (Java method), 333
getActivityType() (Java method), 464
getAdminConfirmPassword() (Java method), 417
getAdminEmail() (Java method), 417
getAdminLogin() (Java method), 417
getAdminPassword() (Java method), 417
getAdvancedSettings(Long) (Java method), 257, 269
getAdvancedSettings(Long, boolean) (Java method), 257, 270
getAfter() (Java method), 421
getAfterTimeUnit() (Java method), 141
getAfterTimeValue() (Java method), 141
getAliases(String) (Java method), 311
getAll() (Java method), 64, 65, 90
getAll(int) (Java method), 65
getAllActivities() (Java method), 487
getAllChannels() (Java method), 483
getAllLogMappings() (Java method), 323
getAllMessages() (Java method), 50
getAllModuleProperties(String, Map) (Java method), 97
getAllTasks() (Java method), 490
getAllTypes() (Java method), 266, 276
getAngularModule() (Java method), 391
getAngularModules() (Java method), 306
getAngularModulesStr() (Java method), 306
getAnnotations() (Java method), 176
getArgs() (Java method), 473, 479
getAutowireCapableBeanFactory() (Java method), 311
getAvailableLocales(HttpServletRequest) (Java method), 412
getBasic() (Java method), 189, 192
getBean() (Java method), 122
getBean(Class) (Java method), 312
getBean(String) (Java method), 311
getBean(String, Class) (Java method), 311
getBean(String, Object) (Java method), 312
getBeanDefinitionCount() (Java method), 312
getBeanDefinitionNames() (Java method), 312
getBeanNamesForType(Class) (Java method), 312
getBeanNamesForType(Class, boolean, boolean) (Java method), 312
getBeansOfType(Class) (Java method), 312
getBeansOfType(Class, boolean, boolean) (Java method), 312
getBeansWithAnnotation(Class) (Java method), 312
getBefore() (Java method), 421
getBody() (Java method), 54
getBrowsing() (Java method), 177
getBrowsingSettings() (Java method), 143
getBundle() (Java method), 93, 307
getBundleApplicationContext() (Java method), 302
getBundleContext() (Java method), 302

getBundleId() (Java method), 391
 getBundleList() (Java method), 303
 getBundleName() (Java method), 359, 370
 getBundleSymbolicName() (Java method), 395
 getBundleVersion() (Java method), 395
 getBytecode() (Java method), 138
 getChannel(String) (Java method), 483
 getChannelIcon(String) (Java method), 483
 getChannelModuleName() (Java method), 441
 getChannelName() (Java method), 476
 getClass(Object, EntityType) (Java method), 267
 getClass(String, EntityType) (Java method), 267
 getClassDefinition(ClassData) (Java method), 227
 getClassDefinition(T) (Java method), 291
 getClasses() (Java method), 485
 getClassForType(String) (Java method), 54
 getClassLoader() (Java method), 312
 getClassName() (Java method), 139, 143, 150, 184
 getClassType() (Java method), 250
 getConfig() (Java method), 413
 getConfigFileChecksum() (Java method), 400, 403
 getConfigLocation() (Java method), 92
 getConfigSource() (Java method), 86, 98
 getConnector(String) (Java method), 69
 getConsumerCount() (Java method), 45
 getContentLength() (Java method), 390
 getContext() (Java method), 322
 getContextLocation(Bundle) (Java method), 326
 getContextPath() (Java method), 325
 getContextPath(Bundle) (Java method), 327
 getCorrectType(AnnotatedElement) (Java method), 293
 getCorrectType(Member) (Java method), 293
 getCouchDbConfig() (Java method), 86
 getCriteria() (Java method), 193
 getCriterion(String) (Java method), 193
 getCriticalMessage() (Java method), 307, 314
 getCriticalMessage(String) (Java method), 413
 getCronDays() (Java method), 330
 getCronExpression() (Java method), 329
 getCss() (Java method), 422
 getCurrentBundleSymbolicName() (Java method), 303
 getCurrentSchemaVersion(String) (Java method), 257, 267, 270
 getCurrentUser() (Java method), 386
 getCustomOperator() (Java method), 197
 getCustomOperators() (Java method), 162
 getData() (Java method), 63
 getDatabaseUrls() (Java method), 419
 getDataNucleusProperties() (Java method), 132
 getDataProviderId() (Java method), 449
 getDataSource(Long, Long, String) (Java method), 467
 getDataSourceObjectValue(String, String, String) (Java method), 488
 getDataSources() (Java method), 467
 getDataSources(Long) (Java method), 468
 getDate() (Java method), 47, 464
 getDateTimes(DateTimes) (Java method), 71
 getDefault() (Java method), 227
 getDefaultName() (Java method), 171, 208
 getDefaultSecurityConfiguration() (Java method), 384
 getDefaultSettings() (Java method), 409
 getDefaultURL() (Java method), 307
 getDefaultValues() (Java method), 152, 174, 187
 getDeleteMode() (Java method), 133, 135, 141
 getDeliveryStatus() (Java method), 109
 getDeliveryStatuses() (Java method), 103
 getDeliveryTime() (Java method), 109
 getDeliveryTimeRange() (Java method), 103
 getDequeueCount() (Java method), 45
 getDescription() (Java method), 171, 208, 429, 432, 434, 440, 460, 474
 getDestination() (Java method), 45
 getDetachedField(T, String) (Java method), 250, 254, 263
 getDetails() (Java method), 159, 160
 getDifferenceOfDatesInYears(Date) (Java method), 73
 getDirection() (Java method), 295
 getDisplayedFields() (Java method), 137, 179
 getDisplayFields(Long) (Java method), 257, 270
 getDisplayName() (Java method), 152, 171, 176, 187, 208, 210, 312, 429, 431–434, 440, 446, 453, 457, 471, 475, 476
 getDraftOwnerUsername() (Java method), 148
 getDrafts() (Java method), 143
 getDriver() (Java method), 83
 getEmail() (Java method), 364, 367, 376
 getEmptyTrash() (Java method), 141
 getEndDate() (Java method), 330, 333
 getEndTime() (Java method), 118, 329, 337
 getEnhancedClassData(String) (Java method), 228
 getEnhancedClasses(boolean) (Java method), 228
 getEnqueueCount() (Java method), 45
 getEntities() (Java method), 267
 getEntitiesWithLookups() (Java method), 257, 270
 getEntity() (Java method), 137, 152, 162, 168, 170
 getEntity(Long) (Java method), 257, 267, 270
 getEntityByClassName(String) (Java method), 257, 270
 getEntityDraft(Long) (Java method), 257, 270
 getEntityDraft(Long, String) (Java method), 257, 270
 getEntityFields(Long) (Java method), 257, 270
 getEntityForEdit(Long) (Java method), 257, 270
 getEntityId() (Java method), 177, 189
 getEntityLookups(Long) (Java method), 258, 270
 getEntityName(String) (Java method), 277
 getEntitySchemaVersion(Object) (Java method), 268
 getEntityVersion() (Java method), 143
 getEnumName() (Java method), 140
 getEnvironment() (Java method), 313, 321
 getEventInfoList() (Java method), 335

getEventKey() (Java method), 433, 444
getEventParameters() (Java method), 434, 478
getExistingConfigFiles() (Java method), 87
getExpirationDate() (Java method), 367
getExpired() (Java method), 381
getExpiredCount() (Java method), 46
getExpression() (Java method), 446
getExternalId() (Java method), 364, 366, 376
getFailureCause() (Java method), 479
getField() (Java method), 157, 159, 160, 221, 295, 452
getField(Long) (Java method), 143
getField(String) (Java method), 143
getFieldIds() (Java method), 201
getFieldKey() (Java method), 445
getFieldName() (Java method), 64, 294
getFieldName(AnnotatedElement) (Java method), 293
getFieldName(Member) (Java method), 293
getFieldNameChanges() (Java method), 148
getFields() (Java method), 143, 163, 168, 170, 205, 453, 464, 471
getFields(Long) (Java method), 258, 271
getFieldType(Field, EntityType) (Java method), 165, 166
getFile(String, FileAccessType) (Java method), 87
getFilename() (Java method), 93, 395
getFilePath() (Java method), 400, 404
getFilterableFields() (Java method), 137, 179
getFilterChainProxy() (Java method), 384
getFilters() (Java method), 448, 468
getForgotViewData(HttpServletRequest) (Java method), 412
getFromAddress() (Java method), 103, 107, 109
getGenericSignature(Field, EntityType) (Java method), 165, 166
getGenericType(AnnotatedElement) (Java method), 293
getGenericType(AnnotatedElement, int) (Java method), 293
getGenericType(Member, int) (Java method), 293
getGetterName(String, CtClass, CtField) (Java method), 224
getHistoryClassData(String) (Java method), 228
getHistoryClassName(String) (Java method), 277
getHistoryForInstance(Object, QueryParams) (Java method), 259, 272
getHost() (Java method), 403
getHour() (Java method), 71
getHours() (Java method), 78
getHttpContext(HttpContext, Bundle) (Java method), 322
getI18n() (Java method), 307
getIcon() (Java method), 390
getId() (Java method), 44, 61, 62, 64, 110, 118, 141, 143, 152, 158–160, 163, 168, 170, 172, 174–177, 184, 189, 192, 195, 197, 199, 201, 209, 313, 361, 373, 460, 470
getId(T) (Java method), 254
getIdentifier() (Java method), 119, 122
getIdentity() (Java method), 57, 61
getImplementationTitle() (Java method), 395
getImplementationVendorID() (Java method), 395
getImplementationVersion() (Java method), 395
getIndexes() (Java method), 177
getInflightCount() (Java method), 46
getInfo() (Java method), 333
getInfrastructure() (Java method), 150
getInstance() (Java method), 292, 303
getInstanceClassName(Object) (Java method), 268
getInstanceId() (Java method), 192
getInstanceId(Object) (Java method), 268
getInstancesFromTrash(String, QueryParams) (Java method), 265, 275
getInterfaceName() (Java method), 150
getInterfaceName(String) (Java method), 228, 277
getJarList() (Java method), 396
getJdoMetadata() (Java method), 248
getJobType() (Java method), 333
getJs() (Java method), 422
getKey() (Java method), 158, 159, 199, 203, 296, 431, 438, 446, 449
getKeyType(String) (Java method), 470, 478
getLang() (Java method), 420
getLangLocalisation(HttpServletRequest) (Java method), 412
getLanguage() (Java method), 400, 404, 417
getLastModificationDate() (Java method), 148
getLastRun() (Java method), 400, 404
getLevel() (Java method), 44, 48
getLib() (Java method), 422
getListenerCount(String) (Java method), 120
getListeners(String) (Java method), 120
getLocale() (Java method), 364, 366, 367, 376
getLocale(String) (Java method), 386
getLocation() (Java method), 87, 391
getLock() (Java method), 301
getLoggers() (Java method), 324
getLoginMode() (Java method), 400, 404, 417
getLoginModeValue() (Java method), 404
getLoginViewData(HttpServletRequest) (Java method), 412
getLogLevel() (Java method), 319
getLogLevels() (Java method), 323
getLogName() (Java method), 319
getLookup() (Java method), 442
getLookupById(Long) (Java method), 143
getLookupByName(Long, String) (Java method), 258, 271
getLookupByName(String) (Java method), 144
getLookupFieldById(Long) (Java method), 163
getLookupFieldByName(String) (Java method), 163
getLookupFields() (Java method), 195, 472

getLookupIds() (Java method), 201
 getLookupName() (Java method), 163, 195
 getLookups() (Java method), 144, 152, 168, 189
 getLookupsDtos() (Java method), 144
 getManipulations() (Java method), 449
 getMax() (Java method), 60
 getMembers(Class, Predicate, Predicate) (Java method), 293
 getMenu(HttpServletRequest) (Java method), 413
 getMessage() (Java method), 103, 107, 110, 464, 473, 479, 480
 getMessage(MessageSourceResolvable, Locale) (Java method), 313
 getMessage(String, Object[], Locale) (Java method), 313
 getMessage(String, Object[], String, Locale) (Java method), 313
 getMessageId() (Java method), 47
 getMessageKey() (Java method), 217
 getMessageMaxRedeliveryCount() (Java method), 124
 getMessageRedeliveryCount() (Java method), 118
 getMessageRedeliveryDelay() (Java method), 124
 getMessages(HttpServletRequest) (Java method), 409
 getMetadata() (Java method), 152, 189
 getMetadata(String) (Java method), 152, 156, 189
 getMetadata(String, String) (Java method), 156
 getMetadataById(Long) (Java method), 153
 getMethod() (Java method), 122
 getMethodName() (Java method), 163, 195
 getMethodsRequired() (Java method), 362, 373
 getMimeType() (Java method), 390
 getMimeType(String) (Java method), 322
 getMin() (Java method), 60
 getMinute() (Java method), 71
 getMinutes() (Java method), 78
 getModificationDate() (Java method), 147
 getModule() (Java method), 93, 139, 144, 184
 getModuleData(String) (Java method), 316
 getModuleDataByAngular(String) (Java method), 316
 getModuleDataByBundle(Bundle) (Java method), 316
 getModuleId(Bundle) (Java method), 327
 getModuleName() (Java method), 44, 48, 307, 391, 397, 432, 440, 476
 getModulePath(BundleName) (Java method), 321
 getModuleProperties(String, String, Properties) (Java method), 98
 getModuleSettings() (Java method), 135
 getModuleVersion() (Java method), 432, 440, 476
 getMotechEvent() (Java method), 329, 330, 337, 339
 getMotechSecurityConfiguration() (Java method), 379
 getName() (Java method), 56, 88, 144, 150, 153, 174, 175, 184, 187, 197, 203, 239, 326, 333, 391, 399, 442, 460, 470
 getNamespace() (Java method), 53, 139, 144, 184, 309
 getNextFireDate() (Java method), 333
 getNotificationRules() (Java method), 51
 getObject() (Java method), 52
 getObjectId() (Java method), 442, 449, 484
 getObjects() (Java method), 470, 486
 getObjectType() (Java method), 53, 449
 getObjectValue() (Java method), 484
 getOpenId() (Java method), 365, 376
 getOpenIdUsers() (Java method), 380, 387
 getOperator() (Java method), 237, 446
 getOptions() (Java method), 203
 getOrder() (Java method), 241, 421, 431, 438, 469
 getOrigin() (Java method), 362, 373
 getOriginalKey() (Java method), 450
 getOriginalRoleName() (Java method), 371
 getOwnerUsername() (Java method), 147
 getPackage(String) (Java method), 277
 getPackageRoot() (Java method), 54
 getPage() (Java method), 112, 241
 getPageNumber() (Java method), 68
 getPageSize() (Java method), 241
 getParameters() (Java method), 118, 331
 getParams() (Java method), 218
 getParent() (Java method), 313
 getParentBeanFactory() (Java method), 313
 getParentEntity() (Java method), 148
 getParentVersion() (Java method), 148
 getPassword() (Java method), 84, 365, 376, 415, 416
 getPasswordConfirmation() (Java method), 416
 getPath() (Java method), 181, 395, 396, 421
 getPattern() (Java method), 362, 373
 getPermissionAccess() (Java method), 362, 373
 getPermissionName() (Java method), 359, 370
 getPermissionNames() (Java method), 360, 371
 getPermissions() (Java method), 378, 384
 getPersistenceManager() (Java method), 251
 getPersistenceManagerFactory() (Java method), 268
 getPlatformSettings() (Java method), 98, 397, 404
 getPrefix() (Java method), 450
 getPrimitive(Class) (Java method), 298
 getPriority() (Java method), 362, 373
 getProperties() (Java method), 94, 135
 getProperties(File) (Java method), 91
 getProperties(String) (Java method), 132, 397
 getPropertyName() (Java method), 230, 234, 235
 getProperty(String) (Java method), 397
 getProperty(String, String) (Java method), 397
 getPropertyDescriptors(Object) (Java method), 296
 getProtocol() (Java method), 362, 373
 getProvider(String) (Java method), 488
 getProviderById(Long) (Java method), 488
 getProviderId() (Java method), 442
 getProviderName() (Java method), 400, 404, 418, 442
 getProviderObject(String) (Java method), 470
 getProviders() (Java method), 488

`getProviderUrl()` (Java method), 401, 404, 418
`getQuartzScheduler()` (Java method), 341
`getQuartzSchedulerFactoryBean()` (Java method), 341
`getQueryParams()` (Java method), 104
`getQueueSize()` (Java method), 46
`getQueueUrl()` (Java method), 418
`getQueueUrls()` (Java method), 419
`getRangeLookupFields()` (Java method), 163
`getRawConfig(String)` (Java method), 397
`getRawConfig(String, String, Resource)` (Java method), 98
`getRecipient()` (Java method), 44
`getRecords()` (Java method), 112
`getRecordsPerPage()` (Java method), 68
`getRedelivered()` (Java method), 47
`getReference()` (Java method), 294
`getRegisteredModules()` (Java method), 316
`getRelatedClass()` (Java method), 167
`getRelatedClassName(Field, EntityType)` (Java method), 166
`getRepeatCount()` (Java method), 337
`getRepeatIntervalInMilliseconds()` (Java method), 337
`getRepository()` (Java method), 150, 254
`getRepositoryName(String)` (Java method), 228, 277
`getResetViewData(HttpServletRequest)` (Java method), 414
`getResource(String)` (Java method), 313, 322, 323
`getResource(String, ClassLoader)` (Java method), 232
`getResourceFileName(Resource)` (Java method), 132, 398
`getResourcePath()` (Java method), 307, 326
`getResourceRootDirectoryPath()` (Java method), 322
`getResources(String)` (Java method), 313
`getResources(String, ClassLoader)` (Java method), 232
`getRestOptions()` (Java method), 144, 177
`getRole(String)` (Java method), 385
`getRoleForAccess()` (Java method), 307, 314
`getRoleName()` (Java method), 360, 372
`getRoles()` (Java method), 365, 366, 376, 378, 385
`getRoles(String)` (Java method), 387
`getRoot()` (Java method), 324
`getRootLogLevel()` (Java method), 323
`getRows()` (Java method), 113
`getRuleById(Long)` (Java method), 379
`getRules()` (Java method), 379
`getRulesByOrigin(String)` (Java method), 379
`getScheduledJobDetailedInfo(JobBasicInfo)` (Java method), 343, 348
`getScheduledJobsBasicInfo()` (Java method), 343, 348
`getScheduledJobTimings(String, String, Date, Date)` (Java method), 343, 348
`getScheduledJobTimingsWithPrefix(String, String, Date, Date)` (Java method), 343, 348
`getSchedulerUrl()` (Java method), 418
`getSchedulerUrls()` (Java method), 420
`getSecurityMembers()` (Java method), 144, 184
`getSecurityMode()` (Java method), 144, 185
`getSecurityRules()` (Java method), 361, 372
`getServerHost()` (Java method), 401, 404
`getServerUrl()` (Java method), 401, 404
`getService(Class)` (Java method), 303
`getServiceImplName(String)` (Java method), 228
`getServiceInterface()` (Java method), 429, 436, 462
`getServiceInterface(String)` (Java method), 228
`getServiceMethod()` (Java method), 429, 436, 462
`getServiceName()` (Java method), 150
`getServiceName(String)` (Java method), 278
`getServletConfig()` (Java method), 53, 309
`getServletContext()` (Java method), 54, 309, 313
`getSetLookupFields()` (Java method), 163
`getSetterName(String)` (Java method), 224
`getSetting(String)` (Java method), 156, 190
`getSetting(String, String)` (Java method), 157
`getSettingAsArray(String)` (Java method), 157
`getSettingAsBoolean(String)` (Java method), 157
`getSettingByName(String)` (Java method), 153
`getSettings()` (Java method), 153, 172, 190
`getSettingsList()` (Java method), 43
`getSettingsURL()` (Java method), 307, 392
`getSimpleName(String)` (Java method), 278
`getSortBy()` (Java method), 68
`getSqlConfig()` (Java method), 86
`getSqlProperties(Properties)` (Java method), 79
`getStackTraceElement()` (Java method), 464
`getStandaloneInstance()` (Java method), 292
`getStandaloneInstance(ClassLoader)` (Java method), 292
`getStartDate()` (Java method), 331, 333, 339
`getStartTime()` (Java method), 329, 337
`getStartupDate()` (Java method), 313
`getStartupFormValidator(StartupForm, MotechUserService)` (Java method), 426
`getStartupViewData(HttpServletRequest)` (Java method), 414
`getState()` (Java method), 392
`getStatus()` (Java method), 333
`getStatusMsgTimeout()` (Java method), 401, 404
`getSteps()` (Java method), 468
`getStringComboboxFields()` (Java method), 144
`getStringValue(String)` (Java method), 326
`getSubject()` (Java method), 104, 107, 110, 118, 331, 429, 434, 475, 476
`getSubMenu()` (Java method), 307
`getSuffixedId()` (Java method), 61
`getSuperClass()` (Java method), 144, 185
`getSupportClasses()` (Java method), 55
`getSupportedLanguages()` (Java method), 409, 412
`getSupportedSchemes()` (Java method), 362, 373
`getSymbolicName()` (Java method), 326, 392, 398
`getSymbolicNames(BundleContext)` (Java method), 492

getTask() (Java method), 464, 488
 getTask(Long) (Java method), 490
 getTaskActivities(Long) (Java method), 487
 getTaskConfig() (Java method), 460
 getTaskErrors() (Java method), 480
 getTaskPossibleErrors() (Java method), 458
 getTenantId() (Java method), 86
 GETTER_PREFIX (Java field), 292
 getText() (Java method), 48, 107
 getTime() (Java method), 331
 getTime(HttpServletRequest) (Java method), 411
 setTimeout() (Java method), 48
 getTimestamp() (Java method), 47
 getTimeUnit() (Java method), 133, 136
 getTimeValue() (Java method), 134, 136
 getToAddress() (Java method), 104, 107, 110
 getToken() (Java method), 367, 416
 getTooltip() (Java method), 153, 187
 getTotal() (Java method), 113
 getTracking() (Java method), 145, 177
 getTrash() (Java method), 324
 getTrashClassData(String) (Java method), 228
 getTrashClassName(String) (Java method), 278
 getTrigger() (Java method), 460
 getTrigger(TaskTriggerInformation) (Java method), 440
 getTriggerParameters() (Java method), 488
 getTriggerTaskEvents() (Java method), 432, 440
 getTriggerValue(String) (Java method), 488
 getType() (Java method), 67, 139, 153, 190, 197, 203, 210, 222, 431, 434, 443, 446, 457, 472
 getType(String) (Java method), 314
 getType(TypeValidation) (Java method), 266, 277
 getTypeClass() (Java method), 172, 209
 getTypeClassName() (Java method), 172
 getTypeSettingOptions() (Java method), 174
 getUIDisplayPosition() (Java method), 153
 getUploadSize() (Java method), 401, 405
 getUptime() (Java method), 411
 getUrl() (Java method), 84, 307, 314
 getUrlResource() (Java method), 87
 getUseGenericParams() (Java method), 163
 getUser(HttpServletRequest) (Java method), 411, 413
 getUser(String) (Java method), 387
 getUserAccess() (Java method), 362, 374
 getUserByEmail(String) (Java method), 387
 getUserLang(HttpServletRequest) (Java method), 412
 getUserLocale(HttpServletRequest) (Java method), 409
 getUsername() (Java method), 365, 367, 376, 415, 420
 getUserName() (Java method), 84, 297, 367
 getUserRoles() (Java method), 297
 getUsers() (Java method), 380, 387
 getValidation() (Java method), 190
 getValidationByName(String) (Java method), 153
 getValidationErrors() (Java method), 460

getValidations() (Java method), 153, 172
 getValidators() (Java method), 426
 getValue() (Java method), 158, 159, 161, 199, 204, 210, 239, 296, 452
 getValue(Field, Object, Object, EntityType, ObjectReference) (Java method), 268
 getValue(KeyInformation) (Java method), 485
 getValue(String) (Java method), 181
 getValueAsString() (Java method), 204
 getValues() (Java method), 140, 181, 463
 getValueType() (Java method), 174, 176
 getVersion() (Java method), 94, 147, 326, 362, 374, 392
 getWrapperForPrimitive(Class) (Java method), 298
 greaterThanOrEqualTo(DateTime, List) (Java method), 73
 GT (Java field), 284, 456
 GT_EQ (Java field), 284

H

handle(MotechEvent) (Java method), 119, 122, 491, 492
 HandlerPredicates (Java class), 484
 handleSecurity(HttpServletRequest, HttpServletResponse) (Java method), 323
 HAS_ANY_MDS_ROLE (Java field), 286
 HAS_DATA_ACCESS (Java field), 286
 HAS_DATA_OR_SCHEMA_ACCESS (Java field), 286
 HAS_SCHEMA_ACCESS (Java field), 286
 HAS_SETTINGS_ACCESS (Java field), 287
 hasActiveAdminUser() (Java method), 387
 hashCode() (Java method), 60, 71, 78, 84, 86, 87, 94, 107, 110, 118, 134, 177, 179, 185, 187, 190, 192, 193, 195, 198, 200, 201, 204, 205, 209, 210, 307, 319, 321, 325, 330, 337, 339, 365, 370, 372, 392, 420, 425, 430, 431, 433–436, 438, 440, 443–445, 447, 448, 450, 452, 453, 457, 460, 463, 465, 468–470, 472, 473, 475, 476, 478, 484
 hasInterface(CtClass, CtClass) (Java method), 225
 hasListener(String) (Java method), 120
 hasManipulations() (Java method), 450
 hasPermission(String) (Java method), 360
 hasPlatformConfigurationFile() (Java method), 87
 hasPrimitive(Class) (Java method), 298
 hasRegisteredChannel() (Java method), 460
 hasRole(String) (Java method), 365, 367
 hasService() (Java method), 430, 436, 463
 hasSettings() (Java method), 172
 hasStatus(int) (Java method), 392
 hasSubject() (Java method), 430, 475, 476
 hasUser(String) (Java method), 387
 hasValidation() (Java method), 172
 hasValidationErrors() (Java method), 460
 HEAD (Java field), 356
 Header (Java class), 421

HeaderOrder (Java class), 422
HISTORY (Java field), 151
HistoryFieldUtil (Java class), 289
HistoryService (Java interface), 258
HistoryServiceImpl (Java class), 272
Hour (Java field), 79
HOURS (Java field), 136
HTTP (Java field), 357
HttpContextFactory (Java class), 322
HTTPMethod (Java enum), 355
HTTPS (Java field), 357
HttpServiceTracker (Java class), 304
HttpServiceTracker(BundleContext, Map) (Java constructor), 304
HttpServiceTrackers (Java class), 304

I

ICON_LOCATIONS (Java field), 390
ID_DISPLAY_FIELD_NAME (Java field), 288
ID_FIELD_NAME (Java field), 288
IdentityProvider (Java interface), 56
Ignore (Java annotation), 126
IllegalLookupException (Java class), 215
IllegalLookupException(String) (Java constructor), 215
IMPLEMENTATION_TITLE (Java field), 394
IMPLEMENTATION_VENDOR_ID (Java field), 394
IMPLEMENTATION_VERSION (Java field), 395
importTask(String) (Java method), 490
incrementMessageRedeliveryCount() (Java method), 118
inDays() (Java method), 78
index(HttpServletRequest) (Java method), 411
indexOfDisplayedField(Long) (Java method), 179
INFO (Java field), 49
info(String, String) (Java method), 51
info(String, String, DateTime) (Java method), 51
inheritsFromCustomClass(Class) (Java method), 225
init() (Java method), 95, 341
initFrameworkServlet() (Java method), 310
initHandler() (Java method), 396
initializeProxyChain() (Java method), 384
initializeSecurityState() (Java method), 254
inMillis() (Java method), 78
inMinutes() (Java method), 78
inRange(DateTime, DateTime, DateTime) (Java method), 73
insertField(Integer, Integer) (Java method), 195
insertField(Integer, Long) (Java method), 195
InSet (Java annotation), 126
INSTALLED (Java field), 393
InstanceSecurityRestriction (Java class), 290
INTEGER (Java field), 193, 207, 279, 458
InvalidTokenException (Java class), 368
InvalidTokenException() (Java constructor), 368
InvalidTokenException(String) (Java constructor), 368
isAbstractClass() (Java method), 145, 185
isActive() (Java method), 362, 365, 367, 374, 376
isActualEntity() (Java method), 145, 147, 149
isAllowCreate() (Java method), 168, 170, 205
isAllowDelete() (Java method), 168, 170, 205
isAllowMultipleSelections() (Java method), 140
isAllowRead() (Java method), 168, 170, 206
isAllowUpdate() (Java method), 168, 170, 206
isAllowUserSupplied() (Java method), 140
isAssignable(Class, List) (Java method), 55
isAssignableFrom(Class, String) (Java method), 232
isAssignableFrom(String, Class) (Java method), 232
isAssignableFrom(String, String) (Java method), 232
isBaseEntity() (Java method), 145
isBefore(Time) (Java method), 71
isBeingTracked(Bundle) (Java method), 304, 317
isBlob() (Java method), 172
isBlueprintEnabled() (Java method), 326
isBootstrapConfigRequired() (Java method), 409
isByCreator() (Java method), 290
isByOwner() (Java method), 290
isCascadeDelete() (Java method), 167
isCascadePersist() (Java method), 167
isCascadeUpdate() (Java method), 167
isChangesMade() (Java method), 149, 182
isClassPersistenceCapable(Class) (Java method), 233
isCombobox() (Java method), 172, 209
isConfigRequired() (Java method), 409
isCreate() (Java method), 181, 201
isDDE() (Java method), 139, 145, 185
isDDEReady(String) (Java method), 228
isDeletable() (Java method), 360, 372
isDelete() (Java method), 201
isDeleted() (Java method), 362, 374
isDraft() (Java method), 145, 149
isEdit() (Java method), 181
isEmpty() (Java method), 290
isEmptyTrash() (Java method), 134, 136
isEnabled() (Java method), 161, 210, 460
isEnabled(boolean) (Java method), 466
isEnum() (Java method), 140
isEnumList() (Java method), 140
isExposedViaRest() (Java method), 153, 163, 195
isFailIfDataNotFound() (Java method), 443
isFailIfNotFound() (Java method), 484
isFile() (Java method), 89
isFileSupported(File) (Java method), 91
isForAdvanced() (Java method), 181
isForField() (Java method), 181
isForSecurity() (Java method), 181
isGeneratePassword() (Java method), 376
isIgnorePastFiresAtStart() (Java method), 330, 331, 337
isInDevelopmentMode() (Java method), 321
isInitialized() (Java method), 309

isInterfaceClass() (Java method), 139
 isLast(Class) (Java method), 290
 isLastEvent() (Java method), 118
 isLessThanADay() (Java method), 79
 isModified() (Java method), 185
 isModuleRegistered(String) (Java method), 316
 isNeedsAttention() (Java method), 308, 315
 isNegationOperator() (Java method), 447
 isValid(ServiceReference) (Java method), 55
 isNullOrEmpty(String) (Java method), 76
 isOneToMany() (Java method), 167
 isOneToOne() (Java method), 167
 isOnOrAfter(DateTime, DateTime) (Java method), 73
 isOnOrBefore(DateTime, DateTime) (Java method), 73
 isOpenId() (Java method), 399
 isOperatorAMethod() (Java method), 237
 isOrderSet() (Java method), 241
 isOutdated() (Java method), 149, 182, 185
 isPagingSet() (Java method), 242
 isParametrized() (Java method), 486
 isPasswordValid(String, String) (Java method), 353
 isPlatformCoreConfigFile(File) (Java method), 91
 isPlatformInitialized() (Java method), 401, 405
 isPrimitive(Class) (Java method), 298
 isPrimitive(String) (Java method), 298
 isPrototype(String) (Java method), 314
 isRangeParam(Field) (Java method), 163
 isRaw() (Java method), 94
 isRead() (Java method), 201
 isReadOnly() (Java method), 43, 153, 164, 185, 190, 195
 isRelationProcessed(String) (Java method), 248
 isRelationship() (Java method), 172, 209
 isRemove() (Java method), 182
 isRepository() (Java method), 399
 isRequired() (Java method), 154, 188, 431, 438
 isRest() (Java method), 363, 374
 isReverse() (Java method), 68
 isSecurityLaunch() (Java method), 420
 isServiceInterfaceRegistered(String) (Java method), 228
 isSetParam(Field) (Java method), 164
 isSingleObjectReturn() (Java method), 164, 196
 isSingleton() (Java method), 53
 isSingleton(String) (Java method), 314
 isString() (Java method), 140
 isStringList() (Java method), 140
 isTracked() (Java method), 154
 isTrashMode() (Java method), 265, 275
 isTypeMatch(String, Class) (Java method), 314
 isUIDisplayable() (Java method), 154
 UIFilterable() (Java method), 154
 isUpdate() (Java method), 202
 isUseGenericParam() (Java method), 198
 isUseOriginalFireTimeAfterMisfire() (Java method), 337
 isValid() (Java method), 430, 459

isValid(ServiceReference) (Java method), 55
 isValid(String) (Java method), 89

J

JAR_FILE_EXTENSION (Java field), 396
 JarGeneratorService (Java interface), 260
 JarGeneratorServiceImpl (Java class), 273
 JarInformation (Java class), 394
 JarInformation(File) (Java constructor), 395
 JarInformationHandler (Java class), 395
 JarInformationHandler(String) (Java constructor), 396
 JavassistBuilder (Java class), 223
 JavassistHelper (Java class), 224
 JavassistLoader (Java class), 227
 JavassistLoader(MDSCClassLoader) (Java constructor), 227
 JOB_GROUP_NAME (Java field), 274, 348
 JOB_ID_KEY (Java field), 343
 JobBasicInfo (Java class), 331
 JobBasicInfo() (Java constructor), 333
 JobBasicInfo(String, String, String, String, String, String, String, String) (Java constructor), 333
 JobDetailedInfo (Java class), 334
 JobDetailedInfo() (Java constructor), 334
 JobDetailedInfo(List) (Java constructor), 335
 JobId (Java class), 335
 JobId(MotechEvent, String) (Java constructor), 335
 JobId(String, String, String) (Java constructor), 335
 JOBTYPES_CRON (Java field), 332
 JOBTYPES_REPEATING (Java field), 332
 JOBTYPES_RUNONCE (Java field), 332
 JodaFormatter (Java class), 75
 JodaFormatter() (Java constructor), 75
 JOIN (Java field), 454

K

KeyEvaluator (Java class), 485
 KeyEvaluator(TaskContext) (Java constructor), 485
 KeyInformation (Java class), 448

L

LANGUAGE (Java field), 82, 417
 le(Time) (Java method), 71
 LESS_DAYS_FROM_NOW (Java field), 456
 LESS_MONTHS_FROM_NOW (Java field), 456
 lessThan(DateTime, List) (Java method), 73
 Level (Java enum), 49
 LIST (Java field), 207, 458
 listEntities() (Java method), 258, 271
 listEntities(boolean) (Java method), 258, 271
 listRawConfigNames(String) (Java method), 98
 listWorkInProgress() (Java method), 258, 271
 load(DigestInputStream) (Java method), 401, 405
 loadBootstrapConfig() (Java method), 92, 99

- [loadBundle\(Bundle\) \(Java method\)](#), [305](#), [393](#)
 - [loadClass\(Bundle, String, ClassPool\) \(Java method\)](#), [226](#)
 - [loadClass\(ClassData\) \(Java method\)](#), [227](#)
 - [loadClass\(T\) \(Java method\)](#), [291](#)
 - [loadConfig\(\) \(Java method\)](#), [100](#)
 - [loadDefaultConfig\(\) \(Java method\)](#), [100](#), [407](#)
 - [Loader \(Java class\)](#), [290](#)
 - [LoaderException \(Java class\)](#), [215](#)
 - [LoaderException\(Throwable\) \(Java constructor\)](#), [215](#)
 - [loadLoggerDbConfiguration\(\) \(Java method\)](#), [305](#)
 - [loadMotechSettings\(\) \(Java method\)](#), [407](#)
 - [loadRoles\(ApplicationContext\) \(Java method\)](#), [389](#)
 - [loadRules\(ApplicationContext\) \(Java method\)](#), [389](#)
 - [LOCAL_DATE \(Java field\)](#), [207](#)
 - [LocaleController \(Java class\)](#), [412](#)
 - [LocaleService \(Java interface\)](#), [409](#)
 - [log\(EmailRecord\) \(Java method\)](#), [114](#)
 - [Log4JBundleLoader \(Java class\)](#), [305](#)
 - [logError\(String\) \(Java method\)](#), [59](#)
 - [logError\(String, Exception\) \(Java method\)](#), [59](#)
 - [logError\(String, Object\) \(Java method\)](#), [59](#)
 - [logger\(\) \(Java method\)](#), [59](#)
 - [Loggers \(Java class\)](#), [324](#)
 - [Loggers\(\) \(Java constructor\)](#), [324](#)
 - [Loggers\(List, LogMapping\) \(Java constructor\)](#), [324](#)
 - [login\(HttpServletRequest\) \(Java method\)](#), [412](#), [413](#)
 - [LOGIN_MODE \(Java field\)](#), [417](#)
 - [LoginController \(Java class\)](#), [412](#)
 - [logInfo\(String, Object\) \(Java method\)](#), [59](#)
 - [LoginForm \(Java class\)](#), [415](#)
 - [LoginMode \(Java class\)](#), [399](#)
 - [LOGINMODE \(Java field\)](#), [82](#)
 - [LogMapping \(Java class\)](#), [318](#)
 - [LogMapping\(\) \(Java constructor\)](#), [318](#)
 - [LogMapping\(String, String\) \(Java constructor\)](#), [318](#)
 - [logObjectIfNotNull\(Object\) \(Java method\)](#), [348](#)
 - [logout\(HttpServletRequest, HttpServletResponse, Authentication\) \(Java method\)](#), [352](#)
 - [LONG \(Java field\)](#), [207](#), [458](#)
 - [Lookup \(Java annotation\)](#), [126](#)
 - [Lookup \(Java class\)](#), [161](#), [451](#)
 - [Lookup\(\) \(Java constructor\)](#), [161](#), [451](#)
 - [Lookup\(LookupDto, List\) \(Java constructor\)](#), [162](#)
 - [Lookup\(String, boolean, boolean, List\) \(Java constructor\)](#), [162](#)
 - [Lookup\(String, boolean, boolean, List, boolean, String\) \(Java constructor\)](#), [162](#)
 - [Lookup\(String, boolean, boolean, List, boolean, String, List, List, Map\) \(Java constructor\)](#), [162](#)
 - [Lookup\(String, boolean, boolean, List, boolean, String, List, List, Map, Map\) \(Java constructor\)](#), [162](#)
 - [Lookup\(String, boolean, boolean, List, Entity\) \(Java constructor\)](#), [162](#)
 - [Lookup\(String, String\) \(Java constructor\)](#), [451](#)
 - [lookup\(String, String, Map\) \(Java method\)](#), [56](#)
 - [lookupCountMethod\(String\) \(Java method\)](#), [291](#)
 - [LookupDto \(Java class\)](#), [194](#)
 - [LookupDto\(\) \(Java constructor\)](#), [194](#)
 - [LookupDto\(Long, String, boolean, boolean, List, boolean, String\) \(Java constructor\)](#), [194](#)
 - [LookupDto\(String, boolean, boolean\) \(Java constructor\)](#), [194](#)
 - [LookupDto\(String, boolean, boolean, List, boolean\) \(Java constructor\)](#), [194](#)
 - [LookupDto\(String, boolean, boolean, List, boolean, String\) \(Java constructor\)](#), [194](#)
 - [LookupExecutionException \(Java class\)](#), [215](#)
 - [LookupExecutionException\(\) \(Java constructor\)](#), [215](#)
 - [LookupExecutionException\(Throwable\) \(Java constructor\)](#), [216](#)
 - [LookupField \(Java annotation\)](#), [126](#)
 - [LookupFieldDto \(Java class\)](#), [197](#)
 - [LookupFieldDto\(\) \(Java constructor\)](#), [197](#)
 - [LookupFieldDto\(Long, String, Type\) \(Java constructor\)](#), [197](#)
 - [LookupFieldDto\(Long, String, Type, String\) \(Java constructor\)](#), [197](#)
 - [LookupFieldDto\(Long, String, Type, String, boolean\) \(Java constructor\)](#), [197](#)
 - [LookupFieldsParameter \(Java class\)](#), [452](#)
 - [LookupFieldsParameter\(\) \(Java constructor\)](#), [452](#)
 - [LookupFieldsParameter\(String, List\) \(Java constructor\)](#), [452](#)
 - [lookupMethod\(String\) \(Java method\)](#), [291](#)
 - [LookupName \(Java class\)](#), [291](#)
 - [LookupNameIsRepeatedException \(Java class\)](#), [216](#)
 - [LookupNameIsRepeatedException\(\) \(Java constructor\)](#), [216](#)
 - [LookupNotFoundException \(Java class\)](#), [216](#)
 - [LookupNotFoundException\(\) \(Java constructor\)](#), [216](#)
 - [LT \(Java field\)](#), [284](#), [456](#)
 - [LT_EQ \(Java field\)](#), [284](#)
- ## M
- [Mail \(Java class\)](#), [106](#)
 - [Mail\(String, String, String, String\) \(Java constructor\)](#), [106](#)
 - [main\(String\[\]\) \(Java method\)](#), [347](#)
 - [MANAGE_PERMISSION_PERMISSION \(Java field\)](#), [356](#)
 - [MANAGE_ROLE_PERMISSION \(Java field\)](#), [357](#)
 - [MANAGE_USER_PERMISSION \(Java field\)](#), [357](#)
 - [ManagementDataProvider \(Java class\)](#), [492](#)
 - [ManagementDataProvider\(TaskDataProviderService\) \(Java constructor\)](#), [492](#)
 - [ManagementDataProvider\(TaskTriggerHandler, TaskDataProviderService\) \(Java constructor\)](#), [492](#)
 - [Manifest \(Java class\)](#), [282](#)
 - [MANIFEST_VERSION \(Java field\)](#), [282](#)

- manipulate(String, String) (Java method), 485
- ManipulationTarget (Java enum), 453
- ManipulationType (Java enum), 454
- MAP (Java field), 207, 458
- MAP_KEY_TYPE (Java field), 283
- MAP_VALUE_TYPE (Java field), 283
- markAsProcessed(ApplicationContext) (Java method), 301
- MATCHES (Java field), 284
- matches(StatusMessage) (Java method), 44
- MatchesProperty (Java class), 238
- MatchesProperty(String, String) (Java constructor), 238
- MAX (Java field), 279
- MAX_REPEAT_COUNT (Java field), 274
- MDS_BUNDLE_NAME (Java field), 280
- MDS_BUNDLE_SYMBOLIC_NAME (Java field), 280
- MDS_COMMON_CONTEXT (Java field), 260
- MDS_DELETE_MODE (Java field), 281
- MDS_EMPTY_TRASH (Java field), 281
- MDS_ENTITIES_CONTEXT (Java field), 260
- MDS_ENTITIES_CONTEXT_TEMPLATE (Java field), 260
- MDS_ENTITIES_NAME (Java field), 280
- MDS_ENTITIES_SYMBOLIC_NAME (Java field), 280
- MDS_MIGRATION_NAME (Java field), 280
- MDS_MIGRATION_SYMBOLIC_NAME (Java field), 280
- MDS_TIME_UNIT (Java field), 281
- MDS_TIME_VALUE (Java field), 282
- MDSClassLoader (Java class), 291
- MDSClassLoader() (Java constructor), 291
- MDSClassLoader(ClassLoader) (Java constructor), 292
- MDSClassLoaderResolver (Java class), 231
- MDSClassLoaderResolver() (Java constructor), 231
- MDSClassLoaderResolver(ClassLoader) (Java constructor), 231
- MDSClassLoaderResolverImpl (Java class), 233
- MDSClassLoaderResolverImpl() (Java constructor), 233
- MDSClassLoaderResolverImpl(ClassLoader) (Java constructor), 233
- MdsConfig (Java class), 132
- MdsConfig() (Java constructor), 132
- MDSConstructor (Java interface), 130
- MDSDataProviderBuilder (Java interface), 131
- MdsException (Java class), 216
- MdsException(String) (Java constructor), 216
- MdsException(String, String) (Java constructor), 217
- MdsException(String, String, Throwable) (Java constructor), 217
- MdsException(String, Throwable) (Java constructor), 217
- MdsJdoAnnotationReader (Java class), 233
- MdsJdoAnnotationReader(MetaDataManager) (Java constructor), 233
- MdsJDOEnhancer (Java class), 211
- MdsJDOEnhancer(Properties, ClassLoader) (Java constructor), 211
- MdsScheduledJob (Java class), 274
- MdsSchedulerException (Java class), 218
- MdsSchedulerException(String, Throwable) (Java constructor), 218
- MdsSchedulerService (Java interface), 262
- MdsSchedulerServiceImpl (Java class), 274
- MdsSchedulerServiceImpl(BundleContext) (Java constructor), 275
- MdsTransactionManager (Java class), 234
- MemberUtil (Java class), 292
- MenuBuilder (Java class), 422
- mergeMaps(Map, Map) (Java method), 58
- mergeWithDefaults(Properties) (Java method), 405
- MetadataDto (Java class), 199
- MetadataDto() (Java constructor), 199
- MetadataDto(Long, String, String) (Java constructor), 199
- MetadataDto(String, String) (Java constructor), 199
- MetadataHolder (Java class), 248
- MetadataKeys (Java class), 283
- MethodHandler (Java class), 485
- MethodHandler(ActionEvent, Map) (Java constructor), 485
- MIN (Java field), 279
- Minute (Java field), 79
- MODIFICATION_DATE_DISPLAY_FIELD_NAME (Java field), 288
- MODIFICATION_DATE_FIELD_NAME (Java field), 289
- ModificationDateValueGenerator (Java class), 234
- MODIFIED_BY_DISPLAY_FIELD_NAME (Java field), 289
- MODIFIED_BY_FIELD_NAME (Java field), 289
- ModifiedByValueGenerator (Java class), 234
- modify(DateTime) (Java method), 231
- modify(String) (Java method), 236
- modify(T) (Java method), 230
- MODULE (Java field), 279
- MODULE_FILE (Java field), 282
- moduleBackToNormal(String) (Java method), 316
- moduleBackToNormal(String, String) (Java method), 316
- ModuleController (Java class), 413
- moduleId() (Java method), 318
- moduleNeedsAttention(String, String) (Java method), 316
- moduleNeedsAttention(String, String, String) (Java method), 316
- ModulePropertiesRecord (Java class), 92
- ModulePropertiesRecord() (Java constructor), 93
- ModulePropertiesRecord(Map, String, String, String, String, boolean) (Java constructor), 93

- ModulePropertiesRecord(Properties, String, String, String, String, boolean) (Java constructor), 93
- ModulePropertiesService (Java interface), 102
- ModuleRegistrationData (Java class), 305
- ModuleRegistrationData() (Java constructor), 305
- ModuleRegistrationData(String, Map) (Java constructor), 306
- ModuleRegistrationData(String, String) (Java constructor), 306
- ModuleRegistrationData(String, String, List, Map) (Java constructor), 306
- MODULES_WITH_SUBMENU (Java field), 315
- MODULES_WITHOUT_SUBMENU (Java field), 315
- MODULES_WITHOUT_UI (Java field), 315
- ModuleSettings (Java class), 133
- Monday (Java field), 70
- MONTHS (Java field), 136
- MORE_DAYS_FROM_NOW (Java field), 456
- MORE_MONTHS_FROM_NOW (Java field), 456
- MOTeCH_EVENT (Java field), 123
- MOTeCH_EVENT_CLASS_NAME (Java field), 82
- MOTeCH_MDS_PROPERTIES (Java field), 260
- MotechAccessVoter (Java class), 351
- MotechBaseDataObject (Java class), 67
- MotechBaseDataObject() (Java constructor), 67
- MotechBaseDataObject(String) (Java constructor), 67
- MotechBaseRepository (Java class), 64
- MotechBaseRepository(Class, CouchDbConnector) (Java constructor), 65
- MotechBasicAuthenticationEntryPoint (Java class), 351
- MotechBasicAuthenticationEntryPoint(SettingsFacade) (Java constructor), 352
- MotechCachingConnectionFactory (Java class), 124
- MotechClassPool (Java class), 227
- MotechConfigurationException (Java class), 80
- MotechConfigurationException(String) (Java constructor), 80
- MotechConfigurationException(String, Exception) (Java constructor), 80
- MotechDataRepository (Java class), 249
- MotechDataRepository(Class) (Java constructor), 249
- MotechDataService (Java interface), 262
- MotechEnumUtils (Java class), 57
- MotechEvent (Java class), 116
- MotechEvent() (Java constructor), 117
- MotechEvent(String) (Java constructor), 117
- MotechEvent(String, Map) (Java constructor), 117
- MotechEventConfig (Java class), 124
- MotechEventHeaderMapper (Java class), 124
- MotechEventTransformer (Java class), 125
- MotechException (Java class), 58
- MotechException(String) (Java constructor), 58
- MotechException(String, Throwable) (Java constructor), 58
- MotechExtenderConfigFactory (Java class), 52
- MotechJsonReader (Java class), 62
- MotechJsonReader() (Java constructor), 62
- MotechJsonReader(FieldNamingStrategy) (Java constructor), 62
- MotechListener (Java annotation), 121
- MotechListenerAbstractProxy (Java class), 121
- MotechListenerAbstractProxy(String, Object, Method) (Java constructor), 121
- MotechListenerEventProxy (Java class), 122
- MotechListenerEventProxy(String, Object, Method) (Java constructor), 122
- MotechListenerNamedParametersProxy (Java class), 123
- MotechListenerNamedParametersProxy(String, Object, Method) (Java constructor), 123
- MotechListenerType (Java enum), 123
- MotechLoginUrlAuthenticationEntryPoint (Java class), 352
- MotechLogoutSuccessHandler (Java class), 352
- MotechMapUtils (Java class), 58
- MotechObject (Java class), 59
- MotechOsgiApplicationContextCreator (Java class), 53
- MotechOsgiConfigurableApplicationContext (Java class), 53
- MotechOsgiConfigurableApplicationContext(String[]) (Java constructor), 53
- MotechOsgiWebApplicationContext (Java class), 309
- MotechOsgiWebApplicationContext() (Java constructor), 309
- MotechParam (Java annotation), 123
- MotechPasswordEncoder (Java class), 352
- MotechPermission (Java class), 358
- MotechPermission() (Java constructor), 359
- MotechPermission(String, String) (Java constructor), 359
- MotechPermissionsDataService (Java interface), 382
- MotechPermissionService (Java interface), 383
- MotechPlatformState (Java enum), 408
- MotechProperties (Java class), 63
- MotechProxyManager (Java class), 384
- MotechRestBasicAuthenticationEntryPoint (Java class), 353
- MotechRestBasicAuthenticationEntryPoint(SettingsFacade) (Java constructor), 353
- MotechRole (Java class), 359
- MotechRole() (Java constructor), 359
- MotechRole(String, List, boolean) (Java constructor), 359
- MotechRolesDataService (Java interface), 382
- MotechRoleService (Java interface), 385
- MotechScheduledJob (Java class), 346
- MotechScheduler (Java class), 346
- MotechSchedulerActionProxyService (Java interface), 341
- MotechSchedulerActionProxyServiceImpl (Java class), 347

- [MotechSchedulerActionProxyServiceImpl\(MotechSchedulerService, ServletRequest, ServletResponse, int, int, int, int, int, int\) \(Java method\), 74](#)
[\(Java constructor\), 347](#)
[MotechSchedulerException \(Java class\), 340](#)
[MotechSchedulerException\(\) \(Java constructor\), 340](#)
[MotechSchedulerException\(String\) \(Java constructor\), 340](#)
[MotechSchedulerException\(String, Throwable\) \(Java constructor\), 340](#)
[MotechSchedulerException\(Throwable\) \(Java constructor\), 340](#)
[MotechSchedulerFactoryBean \(Java class\), 341](#)
[MotechSchedulerFactoryBean\(ApplicationContext, Properties\) \(Java constructor\), 341](#)
[MotechSchedulerService \(Java interface\), 342](#)
[MotechSchedulerServiceImpl \(Java class\), 348](#)
[MotechSchedulerServiceImpl\(MotechSchedulerFactoryBean, SettingsFacade\) \(Java constructor\), 348](#)
[MotechSecurityConfiguration \(Java class\), 360](#)
[MotechSecurityConfiguration\(\) \(Java constructor\), 360](#)
[MotechSecurityConfiguration\(List\) \(Java constructor\), 361](#)
[MotechSettings \(Java interface\), 400](#)
[MotechSuccessHandler \(Java class\), 353](#)
[MotechURL \(Java class\), 402](#)
[MotechURL\(String\) \(Java constructor\), 403](#)
[MotechURLSecurityRule \(Java class\), 361](#)
[MotechURLSecurityRuleDataService \(Java interface\), 382](#)
[MotechURLSecurityService \(Java interface\), 385](#)
[MotechUser \(Java class\), 364](#)
[MotechUser\(\) \(Java constructor\), 364](#)
[MotechUser\(String, String, String, String, List, String, Locale\) \(Java constructor\), 364](#)
[MotechUserProfile \(Java class\), 366](#)
[MotechUserProfile\(MotechUser\) \(Java constructor\), 366](#)
[MotechUsersDataService \(Java interface\), 382](#)
[MotechUserService \(Java interface\), 386](#)
[moveFromTrash\(Object, Object\) \(Java method\), 265, 276](#)
[moveToTrash\(Object, Long\) \(Java method\), 266, 276](#)
- ## N
- [NAME \(Java field\), 180, 279](#)
[NAMED_PARAMETERS \(Java field\), 123](#)
[NAMESPACE \(Java field\), 279](#)
[NanoStopWatch \(Java class\), 59](#)
[NanoStopWatch\(\) \(Java constructor\), 60](#)
[NEED_BOOTSTRAP_CONFIG \(Java field\), 408](#)
[NEED_CONFIG \(Java field\), 408](#)
[NEQ \(Java field\), 284](#)
[newDate\(Date\) \(Java method\), 74](#)
[newDate\(DateTime\) \(Java method\), 74](#)
[newDate\(int, int, int\) \(Java method\), 74](#)
[newDateTime\(Date\) \(Java method\), 74](#)
[newDateTime\(int, int, int\) \(Java method\), 74](#)
[newDateTime\(int, int, int, int, int, int\) \(Java method\), 74](#)
[newDateTime\(int, int, Time\) \(Java method\), 74](#)
[newDateTime\(LocalDate\) \(Java method\), 74](#)
[newDateTime\(LocalDate, int, int, int\) \(Java method\), 74](#)
[newDateTime\(LocalDate, Time\) \(Java method\), 74](#)
[nextApplicableWeekDay\(DateTime, List\) \(Java method\), 74](#)
[nextApplicableWeekDayIncludingFromDate\(DateTime, List\) \(Java method\), 75](#)
[NO \(Java field\), 222](#)
[NO_DB \(Java field\), 408](#)
[NO_METHODS_REQUIRED_EXCEPTION_MESSAGE \(Java field\), 354](#)
[NO_PATTERN_EXCEPTION_MESSAGE \(Java field\), 354](#)
[NO_PROTOCOL_EXCEPTION_MESSAGE \(Java field\), 354](#)
[NO_SECURITY \(Java field\), 358](#)
[NO_SUPPORTED_SCHEMES_EXCEPTION_MESSAGE \(Java field\), 354](#)
[NonAdminUserException \(Java class\), 368](#)
[NonAdminUserException\(String\) \(Java constructor\), 369](#)
[NORMAL_RUN \(Java field\), 408](#)
[NoSuchTypeException \(Java class\), 218](#)
[NoSuchTypeException\(\) \(Java constructor\), 218](#)
[NotificationRule \(Java class\), 44](#)
[NotificationRule\(\) \(Java constructor\), 44](#)
[NotificationRule\(String, ActionType, Level, String\) \(Java constructor\), 44](#)
[NotificationRulesDataService \(Java interface\), 50](#)
[NotInSet \(Java annotation\), 127](#)
[now\(\) \(Java method\), 72, 75, 77](#)
[nowUTC\(\) \(Java method\), 75](#)
[NULL \(Java field\), 474](#)
[NumberPredicate \(Java class\), 293](#)
[NumberPredicate\(Number\) \(Java constructor\), 294](#)
- ## O
- [OATH \(Java field\), 358](#)
[objectEquals\(Long, Long, String\) \(Java method\), 443](#)
[ObjectNotFoundException \(Java class\), 218](#)
[ObjectNotFoundException\(\) \(Java constructor\), 218](#)
[ObjectReadException \(Java class\), 218](#)
[ObjectReadException\(\) \(Java constructor\), 219](#)
[ObjectReadException\(Throwable\) \(Java constructor\), 219](#)
[ObjectReference \(Java class\), 294](#)
[ObjectReference\(String, Object\) \(Java constructor\), 294](#)
[ObjectUpdateException \(Java class\), 219](#)
[ObjectUpdateException\(\) \(Java constructor\), 219](#)
[ObjectUpdateException\(Throwable\) \(Java constructor\), 219](#)
[onAuthenticationSuccess\(HttpServletRequest, HttpServletResponse, Authentication\) \(Java method\), 353](#)

- oneTimeTokenOpenId(String) (Java method), 388
- OneToManyRelationship (Java class), 165
- OneToOneRelationship (Java class), 165
- OPEN_ID (Java field), 358, 399
- OpenIdUserValidator (Java class), 424
- OpenIdUserValidator(UrlValidator) (Java constructor), 424
- Operators (Java class), 283
- OperatorType (Java enum), 455
- OPTIONS (Java field), 356
- Order (Java class), 294
- Order(String) (Java constructor), 295
- Order(String, Direction) (Java constructor), 295
- Order(String, String) (Java constructor), 295
- org.motechproject.admin.domain (package), 43
- org.motechproject.admin.messages (package), 48
- org.motechproject.admin.service (package), 50
- org.motechproject.bundle.extender (package), 52
- org.motechproject.commons.api (package), 54
- org.motechproject.commons.api.json (package), 62
- org.motechproject.commons.api.model (package), 63
- org.motechproject.commons.couchdb.dao (package), 63
- org.motechproject.commons.couchdb.lucene.query (package), 66
- org.motechproject.commons.couchdb.model (package), 67
- org.motechproject.commons.couchdb.query (package), 67
- org.motechproject.commons.couchdb.service (package), 68
- org.motechproject.commons.date (package), 69
- org.motechproject.commons.date.model (package), 69
- org.motechproject.commons.date.util (package), 72
- org.motechproject.commons.date.util.datetime (package), 77
- org.motechproject.commons.date.valueobjects (package), 78
- org.motechproject.commons.sql.service (package), 79
- org.motechproject.config.core (package), 80
- org.motechproject.config.core.constants (package), 80
- org.motechproject.config.core.domain (package), 83
- org.motechproject.config.core.filestore (package), 90
- org.motechproject.config.core.filters (package), 91
- org.motechproject.config.core.service (package), 91
- org.motechproject.config.domain (package), 92
- org.motechproject.config.monitor (package), 95
- org.motechproject.config.service (package), 96
- org.motechproject.email.builder (package), 103
- org.motechproject.email.contract (package), 106
- org.motechproject.email.domain (package), 108
- org.motechproject.email.service (package), 113
- org.motechproject.event (package), 116
- org.motechproject.event.listener (package), 119
- org.motechproject.event.listener.annotations (package), 121
- org.motechproject.event.queue (package), 124
- org.motechproject.mds.annotations (package), 125
- org.motechproject.mds.builder (package), 127
- org.motechproject.mds.config (package), 131
- org.motechproject.mds.domain (package), 137
- org.motechproject.mds.dto (package), 176
- org.motechproject.mds.enhancer (package), 211
- org.motechproject.mds.ex (package), 212
- org.motechproject.mds.filter (package), 221
- org.motechproject.mds.javassist (package), 223
- org.motechproject.mds.jdo (package), 229
- org.motechproject.mds.query (package), 236
- org.motechproject.mds.repository (package), 244
- org.motechproject.mds.service (package), 252
- org.motechproject.mds.service.impl (package), 267
- org.motechproject.mds.util (package), 277
- org.motechproject.osgi.web (package), 300
- org.motechproject.osgi.web.domain (package), 318
- org.motechproject.osgi.web.exception (package), 319
- org.motechproject.osgi.web.ext (package), 320
- org.motechproject.osgi.web.service (package), 323
- org.motechproject.osgi.web.settings (package), 324
- org.motechproject.osgi.web.util (package), 325
- org.motechproject.scheduler.builder (package), 327
- org.motechproject.scheduler.contract (package), 328
- org.motechproject.scheduler.exception (package), 340
- org.motechproject.scheduler.factory (package), 341
- org.motechproject.scheduler.service (package), 341
- org.motechproject.scheduler.service.impl (package), 346
- org.motechproject.security.annotations (package), 350
- org.motechproject.security.authentication (package), 351
- org.motechproject.security.builder (package), 354
- org.motechproject.security.constants (package), 355
- org.motechproject.security.domain (package), 358
- org.motechproject.security.ex (package), 368
- org.motechproject.security.model (package), 370
- org.motechproject.security.repository (package), 377
- org.motechproject.security.service (package), 383
- org.motechproject.server.api (package), 390
- org.motechproject.server.config (package), 396
- org.motechproject.server.config.domain (package), 399
- org.motechproject.server.config.service (package), 406
- org.motechproject.server.startup (package), 408
- org.motechproject.server.ui (package), 409
- org.motechproject.server.ui.ex (package), 410
- org.motechproject.server.web.controller (package), 410
- org.motechproject.server.web.form (package), 415
- org.motechproject.server.web.helper (package), 421
- org.motechproject.server.web.validator (package), 423
- org.motechproject.tasks.annotations (package), 427
- org.motechproject.tasks.contract (package), 428
- org.motechproject.tasks.domain (package), 435

org.motechproject.tasks.ex (package), 479
 org.motechproject.tasks.json (package), 480
 org.motechproject.tasks.repository (package), 482
 org.motechproject.tasks.service (package), 482
 org.motechproject.tasks.util (package), 492
 OsgiDispatcherServlet (Java class), 310
 OsgiDispatcherServlet(BundleContext) (Java constructor), 310
 OsgiDispatcherServlet(BundleContext, ConfigurableWebApplicationContext) (Java constructor), 310
 OsgiWebApplicationContext (Java class), 311
 OsgiWebApplicationContext(ApplicationContext, ConfigurableWebApplicationContext) (Java constructor), 311
 OutboundEventGateway (Java interface), 125
 OWNER (Java field), 297
 OWNER_DISPLAY_FIELD_NAME (Java field), 289
 OWNER_FIELD_NAME (Java field), 289
 OwnerValueGenerator (Java class), 234

P

PACKAGE_JDO (Java field), 261
 Packages (Java class), 284
 PackagesGenerated (Java class), 285
 Pair (Java interface), 296
 PARAM_DISCARDED_MOTECH_EVENT (Java field), 117
 PARAM_INVALID_MOTECH_EVENT (Java field), 117
 PARAM_REDELIVERY_COUNT (Java field), 117
 Parameter (Java class), 456
 Parameter() (Java constructor), 456
 Parameter(String, ParameterType) (Java constructor), 456
 ParameterType (Java enum), 457
 paramsDeclarationForQuery() (Java method), 222
 parse(Object, Class) (Java method), 298
 parse(Object, String) (Java method), 299
 parse(Object, String, ClassLoader) (Java method), 299
 parse(Object, String, String) (Java method), 299
 parse(Object, String, String, ClassLoader) (Java method), 299
 parse(String) (Java method), 172, 450
 parse(String, Locale) (Java method), 76
 parseAll(String) (Java method), 451
 parseDateTime(String) (Java method), 76
 ParseException (Java class), 69
 ParseException(String) (Java constructor), 69
 parseIntToBool(Integer) (Java method), 299
 parseList(List, Class) (Java method), 299
 parseNumber(Object, String) (Java method), 299
 parsePeriod(String) (Java method), 76
 parseString(String, Class) (Java method), 299
 parseString(String, Class, Class) (Java method), 299
 parseString(String, String) (Java method), 299
 parseStringToMap(String) (Java method), 299

parseTime(String, String) (Java method), 71
 PASSWORD (Java field), 415
 PASSWORD_CONFIRMATION (Java field), 415
 PasswordRecoveriesDataService (Java interface), 383
 PasswordRecovery (Java class), 367
 passwordRecoveryRequest(String) (Java method), 388
 PasswordRecoveryService (Java interface), 388
 PAST_7_DAYS (Java field), 222
 PATH (Java field), 180
 PERIOD (Java field), 207
 PermissionDto (Java class), 370
 PermissionDto() (Java constructor), 370
 PermissionDto(MotechPermission) (Java constructor), 370
 PermissionDto(String, String) (Java constructor), 370
 PermissionNames (Java class), 356
 PERSIST (Java field), 279
 PersistedUserValidator (Java class), 424
 PersistedUserValidator(MotechUserService) (Java constructor), 424
 PLATFORM_CORE_CONFIG_FILTER (Java field), 91
 PLATFORM_SETTINGS_CHANGED_EVENT_SUBJECT (Java field), 82
 PlatformSettingsService (Java interface), 407
 PLUSDAYS (Java field), 454
 POSITIVE (Java field), 204
 POST (Java field), 356
 postMessage(StatusMessage) (Java method), 51
 postMessage(String, String, Level) (Java method), 51
 postMessage(String, String, Level, DateTime) (Java method), 51
 postProcessAfterInitialization(Object, String) (Java method), 351, 428
 postProcessBeforeInitialization(Object, String) (Java method), 351, 428
 postProcessWebApplicationContext(ConfigurableWebApplicationContext) (Java method), 310
 prepareHistoryClass(Entity) (Java method), 128
 prepareTrashClass(Entity) (Java method), 128
 processAnnotations(ApplicationContext) (Java method), 351, 428
 processExistingConfigs(List) (Java method), 100
 PROPERTIES_FILE_EXTENSION (Java field), 93
 PropertiesReader (Java class), 90
 Property (Java class), 238
 Property(String, T) (Java constructor), 238
 PropertyBuilder (Java class), 239
 PropertyUtil (Java class), 296
 Protocol (Java enum), 357
 PROVIDER_NAME (Java field), 82, 417
 PROVIDER_URL (Java field), 82, 417
 publishEvent(ApplicationEvent) (Java method), 314
 publishWarningActivity(String, String) (Java method), 488

PUT (Java field), [356](#)

Q

QueryExecution (Java interface), [240](#)

QueryExecutor (Java class), [240](#)

QueryParam (Java class), [67](#)

QueryParam() (Java constructor), [67](#)

QueryParam(int, int, String, boolean) (Java constructor), [68](#)

QueryParams (Java class), [241](#)

QueryParams(Integer, Integer) (Java constructor), [241](#)

QueryParams(Integer, Integer, Order) (Java constructor), [241](#)

QueryParams(Order) (Java constructor), [241](#)

QueryUtil (Java class), [242](#)

queryViewWithKeyList(String, List) (Java method), [65](#)

QUEUE_URL (Java field), [417](#)

QueueMBean (Java class), [45](#)

QueueMBean(String) (Java constructor), [45](#)

QueueMessage (Java class), [46](#)

QueueMessage(String, Boolean, Date) (Java constructor), [47](#)

QueueURLValidator (Java class), [424](#)

QueueURLValidator() (Java constructor), [424](#)

QueueURLValidator(UrlValidator) (Java constructor), [425](#)

R

Range (Java class), [60](#)

RANGE (Java field), [198](#)

Range(T, T) (Java constructor), [60](#)

RangeProperty (Java class), [243](#)

RangeProperty(String, Range) (Java constructor), [243](#)

rawConfigExists(String, String) (Java method), [100](#)

READABLE (Java field), [88](#)

readFromFile(String, Type) (Java method), [62](#)

readFromStream(InputStream, Type) (Java method), [62](#)

readFromString(String, Type) (Java method), [62](#)

readFromString(String, Type, Map) (Java method), [63](#)

rebuildProxyChain() (Java method), [384](#)

RECEIVED (Java field), [108](#)

reconfigure() (Java method), [324](#)

record(Object) (Java method), [259](#), [272](#)

REDIRECT_BOOTSTRAP (Java field), [410](#)

REDIRECT_HOME (Java field), [410](#)

REDIRECT_STARTUP (Java field), [410](#)

refreshAllUsersContextIfActive() (Java method), [390](#)

refreshUserContextIfActive(String) (Java method), [390](#)

regenerateMdsDataBundle(boolean) (Java method), [261](#), [273](#)

regenerateMdsDataBundle(boolean, boolean) (Java method), [261](#), [273](#)

regenerateMdsDataBundleAfterDdeEnhancement(String) (Java method), [261](#), [273](#)

REGEXP (Java field), [279](#)

register(String, String, String, String, List, Locale) (Java method), [387](#)

register(String, String, String, String, List, Locale, boolean, String) (Java method), [387](#)

registerAllProperties() (Java method), [398](#)

registerAllRawConfig() (Java method), [398](#)

registerChannel(ChannelRequest) (Java method), [483](#)

registerChannel(InputStream, String, String) (Java method), [483](#)

registerDDE(String) (Java method), [228](#)

registeredEnums() (Java method), [229](#)

registeredInterfaces() (Java method), [229](#)

registerEnhancedClassData(ClassData) (Java method), [229](#)

registerEnum(String) (Java method), [229](#)

registerHandlerFor(String) (Java method), [491](#), [492](#)

registerHistoryClassData(ClassData) (Java method), [229](#)

registerListener(EventListener, List) (Java method), [120](#)

registerListener(EventListener, String) (Java method), [121](#)

registerModule(ModuleRegistrationData) (Java method), [316](#)

registerProperties(String, Properties) (Java method), [398](#)

registerProvider(InputStream) (Java method), [489](#)

registerProvider(String) (Java method), [489](#)

registerServiceInterface(String, String) (Java method), [229](#)

registersProperties(String, String) (Java method), [100](#)

registerTrashClassData(ClassData) (Java method), [229](#)

registerUserClassLoader(ClassLoader) (Java method), [232](#)

RELATED_CLASS (Java field), [283](#)

RELATED_FIELD (Java field), [283](#)

Relationship (Java class), [166](#)

RelationshipHolder (Java class), [166](#)

RelationshipHolder(ClassData, Field) (Java constructor), [166](#)

RelationshipHolder(Field) (Java constructor), [166](#)

reloadClassLoader() (Java method), [292](#)

reloadMetadata() (Java method), [248](#)

remove(MotechRole) (Java method), [378](#)

remove(MotechSecurityConfiguration) (Java method), [379](#)

remove(MotechUser) (Java method), [380](#)

remove(Object) (Java method), [259](#), [272](#)

remove>PasswordRecovery) (Java method), [381](#)

remove(T) (Java method), [64](#)

REMOVE_INDEX (Java field), [180](#)

removeAction(String) (Java method), [206](#)

removeAll() (Java method), [65](#), [468](#)

removeAll(String, String) (Java method), [65](#)

removeAngularModule(String) (Java method), [308](#)

removeDataProvider(String) (Java method), [491](#)

- removeDataSources() (Java method), 468
- removeDeclaredFieldIfExists(CtClass, String) (Java method), 226
- removeDeclaredMethodIfExists(CtClass, String) (Java method), 226
- removeDefaults(Properties) (Java method), 405
- removedService(ServiceReference, Object) (Java method), 302, 304, 317
- removeField(Integer) (Java method), 196
- removeField(Long) (Java method), 145, 196
- removeField(Number) (Java method), 202, 206
- removeFieldIfExists(CtClass, String) (Java method), 226
- removeFilterableField(Number) (Java method), 179
- removeFilterSets() (Java method), 468
- removeFromProcessed(ApplicationContext) (Java method), 301
- removeIndex(Integer) (Java method), 177
- removeLogger(String) (Java method), 324
- removeLookup(Long) (Java method), 145
- removeLookup(Number) (Java method), 202
- removeMessage(StatusMessage) (Java method), 51
- removeMetadata(Integer) (Java method), 190
- removeMethodIfExists(CtClass, String) (Java method), 226
- removeModuleRecords(List) (Java method), 100
- removeNotificationRule(Long) (Java method), 51
- removeNotificationRule(String) (Java method), 51
- removePermission(String) (Java method), 360
- removeProperties(String, String) (Java method), 100
- removeTrackerFor(Bundle) (Java method), 304, 317
- removeValidationError(String) (Java method), 461
- render(Map, HttpServletRequest, HttpServletResponse) (Java method), 303
- RenderException (Java class), 319
- RenderException(String) (Java constructor), 319
- RenderException(String, Throwable) (Java constructor), 320
- RenderException(Throwable) (Java constructor), 320
- RepeatingJobId (Java class), 335
- RepeatingJobId(MotechEvent) (Java constructor), 336
- RepeatingJobId(String, String) (Java constructor), 336
- RepeatingSchedulableJob (Java class), 336
- RepeatingSchedulableJob() (Java constructor), 336
- RepeatingSchedulableJob(MotechEvent, Date, Date, Integer, Long, boolean) (Java constructor), 336
- RepeatingSchedulableJob(MotechEvent, Date, Date, Long, boolean) (Java constructor), 336
- REPOSITORY (Java field), 285, 399
- REQUIRE (Java field), 205
- RequiredFieldValidator (Java class), 425
- RequiredFieldValidator(String, String) (Java constructor), 425
- requiresConfigurationFiles() (Java method), 101
- requiresFiltering() (Java method), 222
- rescheduleJob(String, String, String) (Java method), 343, 349
- ReservedKeywordException (Java class), 219
- ReservedKeywordException(String) (Java constructor), 219
- reset(ResetForm, HttpServletRequest) (Java method), 414
- ResetController (Java class), 413
- ResetForm (Java class), 415
- ResetFormValidator (Java class), 425
- resetPassword(String, String, String) (Java method), 388
- resetView(HttpServletRequest) (Java method), 414
- RESOLVED (Java field), 393
- resourceFolder() (Java method), 318
- resourceMappings() (Java method), 300, 318
- RestOptions (Java class), 167
- RestOptions() (Java constructor), 167
- RestOptions(Entity) (Java constructor), 167
- RestOptionsDto (Java class), 200
- RestOptionsDto() (Java constructor), 200
- RestOptionsDto(boolean, boolean, boolean, boolean) (Java constructor), 200
- RestrictionProperty (Java class), 243
- RestrictionProperty(InstanceSecurityRestriction, String) (Java constructor), 244
- RETRIEVAL_RETRIES_COUNT (Java field), 274
- retrieve(Entity, String) (Java method), 246
- retrieve(Object) (Java method), 251
- retrieve(String, Object) (Java method), 251, 254, 263
- retrieve(String, Object, InstanceSecurityRestriction) (Java method), 251
- retrieve(String[], Object[]) (Java method), 251
- retrieve(String[], Object[], InstanceSecurityRestriction) (Java method), 251
- retrieveAll() (Java method), 251, 254, 263
- retrieveAll(Entity) (Java method), 247
- retrieveAll(InstanceSecurityRestriction) (Java method), 251
- retrieveAll(List) (Java method), 254, 264
- retrieveAll(List, InstanceSecurityRestriction) (Java method), 252
- retrieveAll(List, QueryParams) (Java method), 254, 264
- retrieveAll(List, QueryParams, InstanceSecurityRestriction) (Java method), 252
- retrieveAll(QueryParams) (Java method), 254, 263
- retrieveAll(QueryParams, InstanceSecurityRestriction) (Java method), 252
- retrieveAll(String) (Java method), 247
- retrieveAll(String, Object) (Java method), 251
- retrieveAll(String, Object, InstanceSecurityRestriction) (Java method), 251
- retrieveAll(String[], Object[], InstanceSecurityRestriction) (Java method), 251
- retrieveAll(String[], Object[], QueryParams, InstanceSecurityRestriction) (Java method), 252

retrieveByClassName(String) (Java method), [245](#), [248](#)
retrieveById(Long) (Java method), [245](#)
retrieveRegisteredBundleNames() (Java method), [101](#)
retrieveUserByCredentials(String, String) (Java method), [387](#)
revertFromTrash(Object, Object) (Java method), [254](#), [264](#)
ROLE_ACCESS_PREFIX (Java field), [358](#)
RoleDto (Java class), [371](#)
RoleDto() (Java constructor), [371](#)
RoleDto(MotechRole) (Java constructor), [371](#)
RoleDto(String, List) (Java constructor), [371](#)
RoleDto(String, List, boolean) (Java constructor), [371](#)
RoleHasUserException (Java class), [369](#)
RoleHasUserException(String) (Java constructor), [369](#)
Roles (Java class), [286](#)
ROLES (Java field), [297](#)
ROOT_LOGGER_NAME (Java field), [323](#)
RunOnceJobId (Java class), [338](#)
RunOnceJobId(MotechEvent) (Java constructor), [339](#)
RunOnceJobId(String, String) (Java constructor), [338](#)
RunOnceSchedulableJob (Java class), [339](#)
RunOnceSchedulableJob(MotechEvent, Date) (Java constructor), [339](#)

S

safeDefineClass(String, byte[]) (Java method), [292](#)
safeGetProperty(Object, String) (Java method), [296](#)
safeRemove(T) (Java method), [64](#), [65](#)
safeScheduleJob(CronSchedulableJob) (Java method), [343](#), [349](#)
safeScheduleRepeatingJob(RepeatingSchedulableJob) (Java method), [344](#), [349](#)
safeScheduleRunOnceJob(RunOnceSchedulableJob) (Java method), [344](#), [349](#)
safeSetProperty(Object, String, Object) (Java method), [296](#)
safeUnscheduleAllJobs(String) (Java method), [344](#), [349](#)
safeUnscheduleJob(String, String) (Java method), [344](#), [349](#)
safeUnscheduleRepeatingJob(String, String) (Java method), [344](#), [349](#)
safeUnscheduleRunOnceJob(String, String) (Java method), [344](#), [349](#)
sameAs(Object) (Java method), [94](#)
Saturday (Java field), [70](#)
save(BootstrapConfig) (Java method), [101](#)
save(Task) (Java method), [491](#)
saveBootstrapConfig(BootstrapConfig) (Java method), [92](#)
saveConfigProperties(String, Properties) (Java method), [398](#)
saveDraftEntityChanges(Long, DraftData) (Java method), [258](#), [271](#)
saveDraftEntityChanges(Long, DraftData, String) (Java method), [258](#), [271](#)
saveModuleSettings(ModuleSettings) (Java method), [136](#)
saveNotificationRules(List) (Java method), [51](#)
savePlatformSetting(String, String) (Java method), [401](#), [405](#)
savePlatformSettings(MotechSettings) (Java method), [101](#), [398](#)
savePlatformSettings(Properties) (Java method), [101](#)
saveRawConfig(String, Resource) (Java method), [398](#)
saveRawConfig(String, String, String, String, InputStream) (Java method), [101](#)
saveRule(NotificationRule) (Java method), [51](#)
scheduleCronJob(String, Map, String, DateTime, DateTime, Boolean) (Java method), [342](#), [347](#)
scheduleDayOfWeekJob(DayOfWeekSchedulableJob) (Java method), [345](#), [349](#)
scheduleDayOfWeekJob(String, Map, DateTime, DateTime, List, DateTime, Boolean) (Java method), [342](#), [347](#)
scheduleEmptyTrashJob() (Java method), [266](#), [276](#)
scheduleJob(CronSchedulableJob) (Java method), [345](#), [349](#)
SCHEDULER_SYMBOLIC_NAME (Java field), [274](#)
scheduleRepeatingJob(long) (Java method), [262](#), [275](#)
scheduleRepeatingJob(RepeatingSchedulableJob) (Java method), [345](#), [349](#)
scheduleRepeatingJob(String, Map, DateTime, DateTime, Integer, Long, Boolean, Boolean) (Java method), [342](#), [347](#)
SchedulerInstantiationException (Java class), [340](#)
SchedulerInstantiationException(Throwable) (Java constructor), [340](#)
SchedulerShutdownException (Java class), [340](#)
SchedulerShutdownException(Throwable) (Java constructor), [341](#)
scheduleRunOnceJob(RunOnceSchedulableJob) (Java method), [345](#), [350](#)
scheduleRunOnceJob(String, Map, DateTime) (Java method), [342](#), [347](#)
SCHEMA_ACCESS (Java field), [287](#)
SchemaGenerator (Java class), [235](#)
SchemaGenerator(JDOPersistenceManagerFactory) (Java constructor), [235](#)
schemaVersion(Class) (Java method), [290](#)
Scheme (Java enum), [357](#)
SECURITY (Java field), [181](#)
SECURITY_REALM_KEY (Java field), [352](#), [353](#)
SecurityAnnotationBeanPostProcessor (Java class), [350](#)
SecurityAnnotationBeanPostProcessor(MotechPermissionService) (Java constructor), [350](#)
SecurityConfigConstants (Java class), [358](#)
SecurityConfigDto (Java class), [372](#)
SecurityConfigException (Java class), [369](#)
SecurityConfigException(String) (Java constructor), [369](#)
SecurityException (Java class), [219](#)

- SecurityException() (Java constructor), 220
- SecurityMode (Java enum), 296
- SecurityRoleLoader (Java class), 389
- SecurityRoleLoader(MotechRoleService) (Java constructor), 389
- SecurityRuleBuilder (Java class), 354
- SecurityRuleDto (Java class), 373
- SecurityRuleLoader (Java class), 389
- SecurityUtil (Java class), 297
- send(Mail) (Java method), 116
- sendEventMessage(MotechEvent) (Java method), 121, 125
- sendLoginInformation(String, String) (Java method), 387
- SENT (Java field), 108
- serialize(ModuleSettings, JsonGenerator, Serializer-Provider) (Java method), 135
- Serializer (Java class), 135
- SERVER_URL (Java field), 82
- ServerLogService (Java interface), 323
- SERVICE (Java field), 285
- SERVICE_IMPL (Java field), 285, 286
- SERVICE_INTERFACE_FIELD (Java field), 481
- SERVICE_METHOD_FIELD (Java field), 481
- SERVICE_NAME (Java field), 55
- ServiceNotFoundException (Java class), 220
- ServiceNotFoundException() (Java constructor), 220
- ServletRegistrationException (Java class), 320
- ServletRegistrationException(String) (Java constructor), 320
- ServletRegistrationException(String, Throwable) (Java constructor), 320
- ServletRegistrationException(Throwable) (Java constructor), 320
- SET (Java field), 198
- setAbstractClass(boolean) (Java method), 145, 185
- setActionParameters(SortedSet) (Java method), 436
- setActions(List) (Java method), 206, 461
- setActionTaskEvents(List) (Java method), 440
- setActionType(ActionType) (Java method), 44
- setActive(boolean) (Java method), 363, 365, 374, 376
- setActivity(String) (Java method), 334
- setActivityType(TaskActivityType) (Java method), 465
- setAdminConfirmPassword(String) (Java method), 418
- setAdminEmail(String) (Java method), 418
- setAdminLogin(String) (Java method), 418
- setAdminPassword(String) (Java method), 418
- setAfter(String) (Java method), 421
- setAfterTimeUnit(TimeUnit) (Java method), 141
- setAfterTimeValue(int) (Java method), 141
- setAllEntities(AllEntities) (Java method), 255, 268, 271
- setAllEntityAudits(AllEntityAudits) (Java method), 271
- setAllEntityDrafts(AllEntityDrafts) (Java method), 271
- setAllowCreate(boolean) (Java method), 168, 170
- setAllowDelete(boolean) (Java method), 168, 170
- setAllowRead(boolean) (Java method), 169, 170
- setAllowUpdate(boolean) (Java method), 169, 170
- setAllSecurityRules(AllMotechSecurityRules) (Java method), 389
- setAllTypes(AllTypes) (Java method), 271, 277
- setAllTypeValidations(AllTypeValidations) (Java method), 277
- setAngularModule(String) (Java method), 392
- setAnnotations(List) (Java method), 176
- setArgs(List) (Java method), 473
- setAuthenticationManager(AuthenticationManager) (Java method), 354
- setBasic(FieldBasicDto) (Java method), 190, 192
- setBasicAuthenticationEntryPoint(AuthenticationEntryPoint) (Java method), 354
- setBefore(String) (Java method), 421
- setBody(Resource) (Java method), 55
- setBody(String) (Java method), 55
- setBrokerUrl(String) (Java method), 124
- setBrowsing(BrowsingSettingsDto) (Java method), 178
- setBundle(Bundle) (Java method), 308
- setBundle(String) (Java method), 94
- setBundleContext(BundleContext) (Java method), 268, 273, 303, 411, 413, 486, 491
- setBundleName(String) (Java method), 359, 370
- setByCreator(boolean) (Java method), 290
- setByOwner(boolean) (Java method), 290
- setChangesMade(boolean) (Java method), 149, 182
- setChannelDecisionManager(ChannelDecisionManager) (Java method), 354
- setChannelName(String) (Java method), 476
- setClassName(String) (Java method), 145, 150, 185
- setConfig(List) (Java method), 133
- setConfigFileChecksum(String) (Java method), 401, 405
- setConfigFiles(List) (Java method), 398
- setConfigLocation(String) (Java method), 54, 310
- setConfigurationScanner(ConfigurationScanner) (Java method), 53
- setConfigurationService(ConfigurationService) (Java method), 398
- setConsumerCount(long) (Java method), 46
- setCoreConfigurationService(CoreConfigurationService) (Java method), 407
- setCountResult(Query) (Java method), 242
- setCreate(boolean) (Java method), 182, 202
- setCriteria(List) (Java method), 194
- setCriticalMessage(String) (Java method), 308, 315
- setCss(List) (Java method), 422
- setCustomOperator(String) (Java method), 198
- setCustomOperators(Map) (Java method), 164
- setDataProviders(Map) (Java method), 491
- setDataService(MotechPermissionsDataService) (Java method), 378

- setDataService(MotechRolesDataService) (Java method), 378
- setDataService(MotechURLSecurityRuleDataService) (Java method), 379
- setDataService(MotechUsersDataService) (Java method), 380
- setDataService>PasswordRecoveriesDataService (Java method), 381
- setDataSources(List) (Java method), 468
- setDate(DateTime) (Java method), 48, 465
- setDefaultName(String) (Java method), 173, 209
- setDefaultURL(String) (Java method), 308
- setDefaultValue(Object) (Java method), 188
- setDefaultValue(String) (Java method), 154, 174
- setDeletable(boolean) (Java method), 360, 372
- setDelete(boolean) (Java method), 202
- setDeleted(boolean) (Java method), 363, 374
- setDeleteMode>DeleteMode (Java method), 134, 142
- setDeleteMode(String) (Java method), 134
- setDequeueCount(long) (Java method), 46
- setDescription(String) (Java method), 173, 209, 440, 461, 475
- setDestination(String) (Java method), 46
- setDetails>TypeSetting (Java method), 159
- setDetails>TypeValidation (Java method), 161
- setDisplayFields(List) (Java method), 179
- setDisplayName(String) (Java method), 154, 173, 176, 188, 209, 211, 441, 447, 453, 457, 472, 475, 477
- setDraftOwnerUsername(String) (Java method), 149
- setDrafts(List) (Java method), 145
- setEdit(boolean) (Java method), 182
- setEmail(String) (Java method), 365, 367, 376
- setEmptyTrash>Boolean (Java method), 134
- setEmptyTrash>boolean (Java method), 142
- setEmptyTrash(String) (Java method), 134
- setEnabled(boolean) (Java method), 161, 211, 461
- setEndDate(String) (Java method), 334
- setEndTime>Date (Java method), 119, 337
- setEnqueueCount(long) (Java method), 46
- setEntity>Entity (Java method), 137, 154, 164, 169, 171
- setEntityId>Long (Java method), 178, 190
- setEntityService>EntityService (Java method), 255
- setEntityVersion>Long (Java method), 146
- setEventAdmin>EventAdmin (Java method), 407
- setEventInfoList(List) (Java method), 335
- setEventKey(String) (Java method), 444
- setEventParameters(List) (Java method), 478
- setExpirationDate(DateTime) (Java method), 367
- setExpiredCount(long) (Java method), 46
- setExposedViaRest>boolean (Java method), 154, 164, 196
- setExpression(String) (Java method), 447
- setExternalId(String) (Java method), 365, 377
- setFailIfDataNotFound>boolean (Java method), 443
- setField>Field (Java method), 158, 159, 161
- setField>Object, String, List (Java method), 289
- setField>String (Java method), 222, 452, 465
- setFieldIds>List (Java method), 202
- setFieldKey>String (Java method), 445
- setFieldName>String (Java method), 294
- setFieldNameChanges>Map (Java method), 149
- setFields>List (Java method), 146, 164, 206, 453, 465, 472
- setFileMonitor>DefaultFileMonitor (Java method), 95
- setFilename>String (Java method), 94
- setFilePath>String (Java method), 401, 405
- setFilterableFields>List (Java method), 179
- setFilters>List (Java method), 448, 468
- setFromAddress>String (Java method), 110
- setGeneratePassword>boolean (Java method), 377
- setHasRegisteredChannel>boolean (Java method), 461
- setHistoryService>HistoryService (Java method), 255, 276
- setHour>Integer (Java method), 72
- setId>Long (Java method), 45, 110, 142, 146, 154, 158, 159, 161, 164, 169, 171, 173–176, 178, 185, 190, 192, 196, 198, 200, 202, 209, 363, 374, 461, 470
- setId>UUID (Java method), 119
- setIgnorePastFiresAtStart>boolean (Java method), 337
- setIndexes>List (Java method), 178
- setInflightCount>long (Java method), 46
- setInfo>String (Java method), 334
- setInstanceId>Long (Java method), 192
- setInterfaceName>String (Java method), 150
- setJobType>String (Java method), 334
- setJs>List (Java method), 422
- setKey>String (Java method), 158, 200, 438, 447
- setLanguage>String (Java method), 401, 405, 418
- setLastEvent>boolean (Java method), 119
- setLastModificationDate(DateTime) (Java method), 149
- setLastRun(DateTime) (Java method), 402, 405
- setLevel>Level (Java method), 45, 48
- setLib>List (Java method), 422
- setLocale>Locale (Java method), 365, 368, 377
- setLocale>String, Locale (Java method), 388
- setLocaleService>LocaleService (Java method), 411, 413
- setLog4jConf>String (Java method), 305
- setLoggers>List (Java method), 325
- setLoginAuthenticationEntryPoint>AuthenticationEntryPoint (Java method), 355
- setLoginMode>String (Java method), 418
- setLoginModeValue>String (Java method), 402, 405
- setLogLevel>String (Java method), 319
- setLogName>String (Java method), 319
- setLookup>Object (Java method), 443
- setLookupFields>List (Java method), 196, 472

- setLookupIds(List) (Java method), 202
- setLookupName(String) (Java method), 164, 196
- setLookups(List) (Java method), 146, 190
- setLookups(Set) (Java method), 154
- setMDSConstructor(MDSConstructor) (Java method), 272
- setMdsConstructor(MDSConstructor) (Java method), 273
- setMdsDataProvider(MDSDataProvider) (Java method), 273
- setMdsSchedulerService(MdsSchedulerService) (Java method), 276
- setMenuBuilder(MenuBuilder) (Java method), 413
- setMessage(String) (Java method), 110, 465, 473
- setMetadata(List) (Java method), 154, 191
- setMetadataHolder(MetadataHolder) (Java method), 273
- setMethodName(String) (Java method), 164, 196
- setMethodsRequired(List) (Java method), 363, 374
- setMinute(Integer) (Java method), 72
- setModificationDate(DateTime) (Java method), 148
- setModified(boolean) (Java method), 186
- setModule(String) (Java method), 94, 146, 186
- setModuleName(String) (Java method), 45, 48, 308, 392, 398, 433, 441, 477
- setModuleVersion(String) (Java method), 433, 441, 477
- setMonitor(EntitiesBundleMonitor) (Java method), 273
- setMotechEvent(MotechEvent) (Java method), 338
- setMotechLogoutHandler(MotechLogoutSuccessHandler) (Java method), 355
- setName(String) (Java method), 146, 155, 174, 175, 186, 188, 198, 204, 334, 443, 461, 471
- setNamespace(String) (Java method), 54, 146, 186, 310
- setNeedsAttention(boolean) (Java method), 308, 315
- setNegationOperator(boolean) (Java method), 447
- setNextFireDate(String) (Java method), 334
- setObjectId(Long) (Java method), 443
- setObjects(List) (Java method), 471
- setOpenId(String) (Java method), 366, 377
- setOpenIdAuthenticationFilter(OpenIdAuthenticationFilter) (Java method), 355
- setOperator(String) (Java method), 447
- setOptions(List) (Java method), 204
- setOrder(int) (Java method), 431
- setOrder(Integer) (Java method), 438, 469
- setOrder(String) (Java method), 421
- setOrigin(String) (Java method), 363, 374
- setOriginalRoleName(String) (Java method), 372
- setOutdated(boolean) (Java method), 183, 186
- setOwnerUsername(String) (Java method), 148
- setPageNumber(int) (Java method), 68
- setParameters(Map) (Java method), 331
- setParentEntity(Entity) (Java method), 149
- setParentVersion(Long) (Java method), 149
- setPassword(String) (Java method), 124, 366, 377, 415, 416
- setPasswordConfirmation(String) (Java method), 416
- setPath(String) (Java method), 422
- setPattern(String) (Java method), 363, 374
- setPermissionAccess(List) (Java method), 363, 375
- setPermissionName(String) (Java method), 359, 371
- setPermissionNames(List) (Java method), 360, 372
- setPersistenceManagerFactory(PersistenceManagerFactory) (Java method), 248, 252, 268
- setPlatformInitialized(boolean) (Java method), 402, 406
- setPlatformSetting(String, String) (Java method), 102
- setPlatformSettings(Map) (Java method), 406
- setPrimary(ClassLoader) (Java method), 232
- setPriority(int) (Java method), 363, 375
- setProperties(EntityDraft, Entity) (Java method), 247
- setProperties(EntityDraft, Entity, String) (Java method), 247
- setProperties(Map) (Java method), 94
- SetProperty (Java class), 244
- SetProperty(String, Set) (Java constructor), 244
- setProperty(String, String) (Java method), 399
- setProtocol(Protocol) (Java method), 363
- setProtocol(String) (Java method), 375
- setProviderId(Long) (Java method), 443
- setProviderName(String) (Java method), 402, 406, 418, 443
- setProviderUrl(String) (Java method), 402, 406, 419
- setProxy(FilterChainProxy) (Java method), 385
- setProxyManager(MotechProxyManager) (Java method), 389
- setQueryParams(Query, QueryParams) (Java method), 242
- setQueueSize(long) (Java method), 46
- setQueueUrl(String) (Java method), 419
- setRangeLookupFields(List) (Java method), 164
- setRaw(boolean) (Java method), 94
- setRawConfigFiles(List) (Java method), 399
- setRead(boolean) (Java method), 202
- setReadOnly(boolean) (Java method), 43, 155, 165, 186, 191, 196
- setRecipient(String) (Java method), 45
- setRecordsPerPage(int) (Java method), 68
- setReference(Object) (Java method), 294
- setRemove(boolean) (Java method), 182
- setRepeatCount(Integer) (Java method), 338
- setRepeatIntervalInMilliseconds(Long) (Java method), 338
- setRepository(MotechDataRepository) (Java method), 255
- setRepository(String) (Java method), 150
- setRequired(boolean) (Java method), 155, 188, 438
- setResourceLoader(ResourceLoader) (Java method), 407
- setResourcePath(String) (Java method), 308
- setRest(boolean) (Java method), 363, 375
- setRestOptions(RestOptions) (Java method), 146

- [setRestOptions\(RestOptionsDto\) \(Java method\), 178](#)
- [setReverse\(boolean\) \(Java method\), 68](#)
- [setRoleForAccess\(List\) \(Java method\), 308, 315](#)
- [setRoleForAccess\(String\) \(Java method\), 308, 315](#)
- [setRoleName\(String\) \(Java method\), 360, 372](#)
- [setRoles\(List\) \(Java method\), 366, 377](#)
- [setRoot\(LogMapping\) \(Java method\), 325](#)
- [setRows\(List\) \(Java method\), 113](#)
- [setRuntimeClassLoader\(ClassLoader\) \(Java method\), 232](#)
- [setSchedulerUrl\(String\) \(Java method\), 419](#)
- [setSecurity\(SecurityMode, List\) \(Java method\), 146](#)
- [setSecurityMembers\(Set\) \(Java method\), 146, 186](#)
- [setSecurityMode\(SecurityMode\) \(Java method\), 146, 186](#)
- [setSecurityRuleBuilder\(SecurityRuleBuilder\) \(Java method\), 385](#)
- [setSecurityRules\(List\) \(Java method\), 361, 373](#)
- [setSecurityRulesDAO\(AllMotechSecurityRules\) \(Java method\), 385](#)
- [setServerUrl\(String\) \(Java method\), 402, 406](#)
- [setServiceInterface\(String\) \(Java method\), 436, 463](#)
- [setServiceMethod\(String\) \(Java method\), 436, 463](#)
- [setServiceName\(String\) \(Java method\), 151](#)
- [setServletConfig\(ServletConfig\) \(Java method\), 54, 310](#)
- [setServletContext\(ServletContext\) \(Java method\), 54, 310](#)
- [setSessionHandler\(SessionHandler\) \(Java method\), 353](#)
- [setSetLookupFields\(List\) \(Java method\), 165](#)
- [setSettings\(List\) \(Java method\), 155, 173, 191](#)
- [setSettingsFacade\(SettingsFacade\) \(Java method\), 355](#)
- [setSettingsList\(List\) \(Java method\), 43](#)
- [setSettingsService\(SettingsService\) \(Java method\), 276](#)
- [setSettingsURL\(String\) \(Java method\), 308, 392](#)
- [setSingleObjectReturn\(boolean\) \(Java method\), 165, 196](#)
- [setSortBy\(String\) \(Java method\), 68](#)
- [setSourceInstance\(DateTimeSource\) \(Java method\), 72](#)
- [setStackTraceElement\(String\) \(Java method\), 465](#)
- [setStartDate\(String\) \(Java method\), 334](#)
- [setStartTime\(Date\) \(Java method\), 338](#)
- [setStartupFormValidatorFactory\(StartupFormValidatorFactory\) \(Java method\), 414](#)
- [setStartupManager\(StartupManager\) \(Java method\), 411](#)
- [setStatus\(String\) \(Java method\), 334](#)
- [setStatusMsgTimeout\(String\) \(Java method\), 402, 406](#)
- [setSubject\(String\) \(Java method\), 111, 331, 475, 477](#)
- [setSubMenu\(Map\) \(Java method\), 309](#)
- [setSuperClass\(String\) \(Java method\), 147, 186](#)
- [setSupportedSchemes\(List\) \(Java method\), 364, 375](#)
- [setTask\(Long\) \(Java method\), 465](#)
- [setTaskConfig\(TaskConfig\) \(Java method\), 461](#)
- [setTaskPossibleErrors\(String\) \(Java method\), 459](#)
- [SETTER_PREFIX \(Java field\), 293](#)
- [setText\(String\) \(Java method\), 48](#)
- [setTimeout\(DateTime\) \(Java method\), 48](#)
- [setTimeUnit\(String\) \(Java method\), 134](#)
- [setTimeUnit\(TimeUnit\) \(Java method\), 134](#)
- [setTimeValue\(Integer\) \(Java method\), 134](#)
- [setTimeValue\(String\) \(Java method\), 134](#)
- [setTimeZone\(DateTime\) \(Java method\), 75](#)
- [setTimeZoneUTC\(DateTime\) \(Java method\), 75](#)
- [SettingDto \(Java class\), 203](#)
- [SettingDto\(\) \(Java constructor\), 203](#)
- [SettingDto\(String, Object\) \(Java constructor\), 203](#)
- [SettingDto\(String, Object, TypeDto, SettingOptions\) \(Java constructor\), 203](#)
- [SettingOptions \(Java enum\), 204](#)
- [Settings \(Java class\), 287](#)
- [SETTINGS \(Java field\), 82](#)
- [SETTINGS_ACCESS \(Java field\), 287](#)
- [SETTINGS_CACHE_NAME \(Java field\), 96](#)
- [SETTINGS_FILE_NAME \(Java field\), 83](#)
- [SettingsDto \(Java class\), 458](#)
- [SettingService \(Java interface\), 407](#)
- [SettingsFacade \(Java class\), 396](#)
- [SettingsRecord \(Java class\), 403](#)
- [SettingsRecord\(\) \(Java constructor\), 403](#)
- [SettingsService \(Java interface\), 135](#)
- [setToAddress\(String\) \(Java method\), 111](#)
- [setToken\(String\) \(Java method\), 368, 416](#)
- [setTooltip\(String\) \(Java method\), 155, 188](#)
- [setTotal\(Integer\) \(Java method\), 113](#)
- [setTracked\(boolean\) \(Java method\), 155](#)
- [setTracking\(Tracking\) \(Java method\), 147](#)
- [setTracking\(TrackingDto\) \(Java method\), 178](#)
- [setTransactionManager\(JdoTransactionManager\) \(Java method\), 255](#)
- [setTrash\(List\) \(Java method\), 325](#)
- [setTrashFlag\(Object, Object, boolean\) \(Java method\), 259, 272](#)
- [setTrashService\(TrashService\) \(Java method\), 255](#)
- [setTrigger\(TaskTriggerInformation\) \(Java method\), 461](#)
- [setTriggerTaskEvents\(List\) \(Java method\), 441](#)
- [setType\(FilterType\) \(Java method\), 222](#)
- [setType\(ParameterType\) \(Java method\), 447, 457](#)
- [setType\(String\) \(Java method\), 67, 443, 472](#)
- [setType\(Type\) \(Java method\), 155, 198](#)
- [setType\(TypeDto\) \(Java method\), 191, 204, 211](#)
- [setTypeClass\(Class\) \(Java method\), 173](#)
- [setTypeClass\(String\) \(Java method\), 209](#)
- [setTypeSettingOptions\(List\) \(Java method\), 174](#)
- [setUIDisplayable\(boolean\) \(Java method\), 155](#)
- [setUIDisplayPosition\(Long\) \(Java method\), 155](#)
- [setUIFilterable\(boolean\) \(Java method\), 155](#)
- [setUiFrameworkService\(UiFrameworkService\) \(Java method\), 413](#)
- [setUpdate\(boolean\) \(Java method\), 202](#)
- [setUploadSize\(String\) \(Java method\), 402, 406](#)
- [setUrl\(String\) \(Java method\), 309, 315](#)
- [setUseGenericParam\(boolean\) \(Java method\), 198](#)
- [setUseGenericParams\(Map\) \(Java method\), 165](#)

- [setUseOriginalFireTimeAfterMisfire\(boolean\) \(Java method\)](#), 338
[setUserAccess\(List\) \(Java method\)](#), 364, 375
[setUserLang\(HttpServletRequest, HttpServletResponse, String, String, String\) \(Java method\)](#), 412
[setUserLocale\(HttpServletRequest, HttpServletResponse, Locale\) \(Java method\)](#), 410
[setUserName\(String\) \(Java method\)](#), 366, 377, 415
[setUsername\(String\) \(Java method\)](#), 124, 368
[setUsernamePasswordAuthenticationFilter\(UsernamePasswordAuthenticationFilter\) \(Java method\)](#), 355
[setValidation\(FieldValidationDto\) \(Java method\)](#), 191
[setValidationErrors\(Set\) \(Java method\)](#), 461
[setValidations\(List\) \(Java method\)](#), 155, 173
[setValue\(Object\) \(Java method\)](#), 204, 211
[setValue\(String\) \(Java method\)](#), 158, 160, 161, 200, 452
[setValues\(Map\) \(Java method\)](#), 182, 463
[setValueType\(Type\) \(Java method\)](#), 174, 176
[setVelocityEngine\(VelocityEngine\) \(Java method\)](#), 274
[setVersion\(Long\) \(Java method\)](#), 148
[setVersion\(String\) \(Java method\)](#), 94, 364, 375
[shouldIgnoreThisProperty\(\) \(Java method\)](#), 236, 239, 243, 244
[shutdown\(\) \(Java method\)](#), 341
[singleResult\(List\) \(Java method\)](#), 66
[SMS \(Java field\)](#), 49
[SPLIT \(Java field\)](#), 454
[SQL_DRIVER \(Java field\)](#), 85
[SQL_PASSWORD \(Java field\)](#), 85
[SQL_URL \(Java field\)](#), 85
[SQL_USER \(Java field\)](#), 85
[SQLDBConfig \(Java class\)](#), 89
[SQLDBConfig\(String, String, String, String\) \(Java constructor\)](#), 89
[SqlDBManager \(Java interface\)](#), 79
[STANDARD \(Java field\)](#), 151
[start\(\) \(Java method\)](#), 60, 304, 317
[start\(BundleContext\) \(Java method\)](#), 300, 301, 318
[STARTING \(Java field\)](#), 393
[STARTS_WITH \(Java field\)](#), 284
[STARTSWITH \(Java field\)](#), 456
[STARTUP \(Java field\)](#), 408
[startup\(\) \(Java method\)](#), 409, 414
[StartupController \(Java class\)](#), 414
[StartupForm \(Java class\)](#), 416
[StartupFormValidator \(Java class\)](#), 426
[StartupFormValidator\(\) \(Java constructor\)](#), 426
[StartupFormValidatorFactory \(Java class\)](#), 426
[StartupManager \(Java class\)](#), 408
[StartupSuggestionsForm \(Java class\)](#), 419
[StartupSuggestionsForm\(\) \(Java constructor\)](#), 419
[State \(Java enum\)](#), 392
[status\(\) \(Java method\)](#), 415
[STATUS_BLOCKED \(Java field\)](#), 332
[STATUS_ERROR \(Java field\)](#), 332
[STATUS_MSG_TIMEOUT \(Java field\)](#), 83
[STATUS_OK \(Java field\)](#), 332
[STATUS_PAUSED \(Java field\)](#), 332
[StatusController \(Java class\)](#), 414
[StatusMessage \(Java class\)](#), 47
[StatusMessage\(\) \(Java constructor\)](#), 47
[StatusMessage\(String, String, Level\) \(Java constructor\)](#), 47
[StatusMessage\(String, String, Level, DateTime\) \(Java constructor\)](#), 47
[StatusMessagesDataService \(Java interface\)](#), 52
[StatusMessageService \(Java interface\)](#), 50
[stop\(\) \(Java method\)](#), 95
[stop\(BundleContext\) \(Java method\)](#), 300, 301, 318
[STOPPING \(Java field\)](#), 393
[STRING \(Java field\)](#), 193, 208, 453
[STRING_MAX_LENGTH \(Java field\)](#), 287
[StringUtil \(Java class\)](#), 76
[SUBJECT_FIELD \(Java field\)](#), 481
[submenuBackToNormal\(String\) \(Java method\)](#), 309
[SubmenuInfo \(Java class\)](#), 314
[SubmenuInfo\(String\) \(Java constructor\)](#), 314
[subMenuNeedsAttention\(String\) \(Java method\)](#), 309
[submitForm\(StartupForm\) \(Java method\)](#), 414
[SUBSTRING \(Java field\)](#), 454
[SUCCESS \(Java field\)](#), 466
[SUFFIX_REPEATJOBID \(Java field\)](#), 336
[SUFFIX_RUNONCEJOBID \(Java field\)](#), 338
[suggestActivemqUrl\(\) \(Java method\)](#), 423
[SuggestionHelper \(Java class\)](#), 423
[Sunday \(Java field\)](#), 70
[SUPPORTED_FILE_EXTNS \(Java field\)](#), 83
[supports\(Class\) \(Java method\)](#), 351
[supports\(ConfigAttribute\) \(Java method\)](#), 351
[supports\(String\) \(Java method\)](#), 55, 56
[SYMBOLIC_NAME_PREFIX \(Java field\)](#), 280
[SYMBOLIC_NAME_SUFFIX \(Java field\)](#), 282
[SystemIdentityProvider \(Java class\)](#), 61
- ## T
- [Task \(Java class\)](#), 459
[Task\(\) \(Java constructor\)](#), 459
[Task\(String, TaskTriggerInformation, List\) \(Java constructor\)](#), 459
[Task\(String, TaskTriggerInformation, List, TaskConfig, boolean, boolean\) \(Java constructor\)](#), 459
[TaskAction \(Java annotation\)](#), 427
[TaskActionExecutor \(Java class\)](#), 486
[TaskActionExecutor\(TaskService, TaskActivityService, EventRelay\) \(Java constructor\)](#), 486
[TaskActionInformation \(Java class\)](#), 462
[TaskActionInformation\(\) \(Java constructor\)](#), 462

`TaskActionInformation(String, String, String, String, String)` (Java constructor), [462](#)
`TaskActionInformation(String, String, String, String, String, Map)` (Java constructor), [462](#)
`TaskActionInformation(String, String, String, String, String, String)` (Java constructor), [462](#)
`TaskActionInformation(String, String, String, String, String, String, String, Map)` (Java constructor), [462](#)
`TaskActionParam` (Java annotation), [428](#)
`TaskActivitiesDataService` (Java interface), [482](#)
`TaskActivity` (Java class), [463](#)
`TaskActivity()` (Java constructor), [463](#)
`TaskActivity(String, List, Long, TaskActivityType)` (Java constructor), [464](#)
`TaskActivity(String, List, Long, TaskActivityType, String)` (Java constructor), [464](#)
`TaskActivity(String, Long, TaskActivityType)` (Java constructor), [463](#)
`TaskActivity(String, String, Long, TaskActivityType)` (Java constructor), [463](#)
`TaskActivityService` (Java interface), [486](#)
`TaskActivityType` (Java enum), [465](#)
`TaskAnnotationBeanPostProcessor` (Java class), [428](#)
`TaskAnnotationBeanPostProcessor(BundleContext, ChannelService)` (Java constructor), [428](#)
`TaskBuilder` (Java class), [466](#)
`TaskBuilder()` (Java constructor), [466](#)
`TaskChannel` (Java annotation), [428](#)
`TaskConfig` (Java class), [467](#)
`TaskConfigDeserializer` (Java class), [481](#)
`TaskConfigStep` (Java class), [468](#)
`TaskContext` (Java class), [487](#)
`TaskContext(Task, MotechEvent, TaskActivityService)` (Java constructor), [487](#)
`TaskDataProvider` (Java class), [469](#)
`TaskDataProvider()` (Java constructor), [469](#)
`TaskDataProvider(String, List)` (Java constructor), [469](#)
`TaskDataProviderObject` (Java class), [471](#)
`TaskDataProviderObject()` (Java constructor), [471](#)
`TaskDataProviderObject(String, String, List, List)` (Java constructor), [471](#)
`TaskDataProviderService` (Java interface), [488](#)
`TaskDeserializer` (Java class), [482](#)
`TaskError` (Java class), [472](#)
`TaskError()` (Java constructor), [472](#)
`TaskError(String, String)` (Java constructor), [473](#)
`TaskError(TaskErrorType, String)` (Java constructor), [472](#)
`TaskErrorType` (Java enum), [473](#)
`TaskEvent` (Java class), [474](#)
`TaskEvent()` (Java constructor), [474](#)
`TaskEvent(String, String, String)` (Java constructor), [474](#)
`TaskEventInformation` (Java class), [475](#)
`TaskEventInformation()` (Java constructor), [475](#)
`TaskEventInformation(String, String, String, String, String)` (Java constructor), [476](#)
`TaskFilterExecutor` (Java class), [489](#)
`TaskHandlerException` (Java class), [479](#)
`TaskHandlerException(TaskFailureCause, String)` (Java constructor), [479](#)
`TaskHandlerException(TaskFailureCause, String, String)` (Java constructor), [479](#)
`TaskHandlerException(TaskFailureCause, String, Throwable, String)` (Java constructor), [479](#)
`TaskInitializer` (Java class), [489](#)
`TaskInitializer(TaskContext)` (Java constructor), [489](#)
`TaskNotFoundException` (Java class), [480](#)
`TaskNotFoundException(Long)` (Java constructor), [480](#)
`TasksDataService` (Java interface), [482](#)
`TaskService` (Java interface), [490](#)
`TaskTriggerHandler` (Java class), [491](#)
`TaskTriggerHandler(TaskService, TaskActivityService, EventListenerRegistryService, EventRelay, SettingsFacade)` (Java constructor), [491](#)
`TaskTriggerInformation` (Java class), [477](#)
`TaskTriggerInformation()` (Java constructor), [477](#)
`TaskTriggerInformation(String, String, String, String, String)` (Java constructor), [477](#)
`TaskTriggerInformation(TaskTriggerInformation)` (Java constructor), [477](#)
`Tenant` (Java class), [61](#)
`Tenant(TenantIdentity)` (Java constructor), [61](#)
`TENANT_ID` (Java field), [85](#)
`TenantIdentity` (Java class), [61](#)
`TenantIdentity()` (Java constructor), [62](#)
`TenantIdentity(IdentityProvider)` (Java constructor), [62](#)
`TEXTAREA` (Java field), [458](#)
`THIS_MONTH` (Java field), [223](#)
`THIS_YEAR` (Java field), [223](#)
`Thursday` (Java field), [70](#)
`Time` (Java class), [70](#)
`TIME` (Java field), [208](#), [458](#)
`Time()` (Java constructor), [70](#)
`time(DateTime)` (Java method), [75](#)
`Time(int, int)` (Java constructor), [70](#)
`Time(Integer, Integer)` (Java constructor), [70](#)
`Time(LocalTime)` (Java constructor), [70](#)
`Time(String)` (Java constructor), [71](#)
`timeStr()` (Java method), [72](#)
`TimeTypeConverter` (Java class), [235](#)
`TimeUnit` (Java enum), [136](#)
`timeZone()` (Java method), [72](#), [77](#)
`toClassPath(Class)` (Java method), [226](#)
`toClassPath(String)` (Java method), [226](#)
`toClassPath(String, boolean)` (Java method), [226](#)
`toDatastoreType(Time)` (Java method), [235](#)
`TODAY` (Java field), [223](#)
`today()` (Java method), [72](#), [75](#), [77](#)

- toDto() (Java method), 137, 147, 149, 156, 158, 160, 161, 165, 169, 171, 173
 - toEnumSet(Class, Set) (Java method), 57
 - toEnumSet(Class, String) (Java method), 57
 - toGenericParam(Class) (Java method), 226
 - toGenericParam(String) (Java method), 226
 - toJSON() (Java method), 55, 56
 - TOLOWER (Java field), 454
 - toMemberType(String) (Java method), 235
 - tomorrow() (Java method), 75
 - toMotechEvent() (Java method), 441
 - toRange(Object, String) (Java method), 300
 - toResource() (Java method), 88
 - toSet(Object, String) (Java method), 300
 - toString() (Java method), 72, 84, 86, 88, 89, 95, 108, 111, 113, 119, 135, 139, 176, 178, 180, 186, 188, 191, 192, 194, 196, 200, 202, 204, 206, 209, 211, 295, 319, 330, 335, 338, 340, 359, 366, 371, 372, 403, 420, 430, 431, 433–435, 437, 439, 441, 444, 445, 447, 448, 453, 457, 461, 463, 465, 468, 469, 471–473, 475, 477, 478
 - toString(Set) (Java method), 57
 - toStringSet(Set) (Java method), 58
 - TOUPPER (Java field), 454
 - TRACE (Java field), 356
 - Tracking (Java class), 169
 - Tracking() (Java constructor), 169
 - Tracking(Entity) (Java constructor), 169
 - TrackingDto (Java class), 205
 - TransactionalMotechDataService (Java class), 264
 - transform(MotechEvent) (Java method), 125
 - TRASH (Java field), 132, 151
 - TrashClassNotFoundException (Java class), 220
 - TrashClassNotFoundException(String) (Java constructor), 220
 - trashFlag(Class) (Java method), 290
 - TrashService (Java interface), 264
 - TrashServiceImpl (Java class), 275
 - TRIGGER_PREFIX (Java field), 449
 - TriggerEvent (Java class), 477
 - TriggerEvent() (Java constructor), 478
 - TriggerEvent(String, String, String, List) (Java constructor), 478
 - TriggerEvent(TriggerEventRequest) (Java constructor), 478
 - TriggerEventRequest (Java class), 434
 - TriggerEventRequest(String, String, String, List) (Java constructor), 434
 - TriggerHandler (Java interface), 492
 - TriggerNotFoundException (Java class), 480
 - TriggerNotFoundException(String) (Java constructor), 480
 - trimTrashHistorySuffix(String) (Java method), 278
 - TRUE (Java field), 289
 - Tuesday (Java field), 70
 - Type (Java class), 171
 - Type (Java enum), 198
 - Type() (Java constructor), 171
 - Type(Class) (Java constructor), 171
 - Type(String, String, Class) (Java constructor), 171
 - TYPE_CLASS (Java field), 181
 - TypeAlreadyExistsException (Java class), 220
 - TypeAlreadyExistsException() (Java constructor), 220
 - TypeDto (Java class), 206
 - TypeDto() (Java constructor), 208
 - TypeDto(Long, String, String, String, String) (Java constructor), 208
 - TypeDto(String, String, String, String) (Java constructor), 208
 - TypeHelper (Java class), 297
 - TypeNotFoundException (Java class), 220
 - TypeNotFoundException(String) (Java constructor), 221
 - TypeService (Java interface), 266
 - TypeServiceImpl (Java class), 276
 - TypeSetting (Java class), 173
 - TypeSetting() (Java constructor), 173
 - TypeSetting(String) (Java constructor), 174
 - TypeSettingOption (Java class), 175
 - TypeSettingOption(String) (Java constructor), 175
 - TypeValidation (Java class), 175
 - TypeValidation() (Java constructor), 175
 - TypeValidation(String, Type) (Java constructor), 175
 - TypeValidationAlreadyExistsException (Java class), 221
 - TypeValidationAlreadyExistsException() (Java constructor), 221
- ## U
- UI (Java field), 88
 - UIDisplayable (Java annotation), 127
 - UIFilterable (Java annotation), 127
 - UIFrameworkService (Java interface), 315
 - UiHttpContext (Java class), 322
 - UiHttpContext(HttpContext) (Java constructor), 322
 - UIServiceTracker (Java class), 317
 - UIServiceTracker(BundleContext, ModuleRegistrationData) (Java constructor), 317
 - UIServiceTracker(BundleContextWrapper, ModuleRegistrationData) (Java constructor), 317
 - UIServiceTrackers (Java class), 317
 - unbind(Object, Map) (Java method), 493
 - underscore() (Java method), 321
 - UNICODE (Java field), 458
 - UNINSTALLED (Java field), 393
 - UNKNOWN (Java field), 132, 136, 393, 455, 458
 - unregister() (Java method), 304
 - unregisterEnhancedData(String) (Java method), 229
 - unregisterModule(String) (Java method), 316
 - unregisterProperties(String) (Java method), 399

- `unscheduleAllJobs(String)` (Java method), [345](#), [350](#)
 - `unscheduleJob(JobId)` (Java method), [346](#), [350](#)
 - `unscheduleJob(String, String)` (Java method), [345](#), [350](#)
 - `unscheduleJobs(String)` (Java method), [342](#), [347](#)
 - `unscheduleRepeatingJob()` (Java method), [262](#), [275](#)
 - `unscheduleRepeatingJob(String, String)` (Java method), [346](#), [350](#)
 - `unscheduleRunOnceJob(String, String)` (Java method), [346](#), [350](#)
 - `unsetPrimary()` (Java method), [232](#)
 - `unwrap()` (Java method), [236](#), [239](#), [243](#), [244](#)
 - `UPDATE` (Java field), [279](#)
 - `update(EntityDraft)` (Java method), [247](#)
 - `update(FieldDto)` (Java method), [156](#)
 - `update(LookupDto, List)` (Java method), [165](#)
 - `update(MetadataDto)` (Java method), [158](#)
 - `update(MotechRole)` (Java method), [378](#)
 - `update(MotechUser)` (Java method), [380](#)
 - `update>PasswordRecovery)` (Java method), [381](#)
 - `update(RestOptionsDto)` (Java method), [169](#)
 - `update(T)` (Java method), [64](#), [252](#), [255](#), [264](#)
 - `updateAdvancedSetting(AdvancedSettingsDto)` (Java method), [147](#)
 - `updateComboboxValues(Long, Map)` (Java method), [258](#), [272](#)
 - `updateConfigLocation(String)` (Java method), [102](#)
 - `updateDraft(Long)` (Java method), [258](#), [272](#)
 - `updateFields(Long, Map)` (Java method), [131](#)
 - `updateFileMonitor()` (Java method), [95](#)
 - `updateFromDraft(EntityDraft)` (Java method), [147](#)
 - `updateFromProperties(Properties)` (Java method), [402](#), [406](#)
 - `updatePropertiesAfterReinstallation(String, String, String, String, Properties, Properties)` (Java method), [102](#)
 - `updateRole(RoleDto)` (Java method), [385](#)
 - `updateScheduledJob(MotechEvent)` (Java method), [346](#), [350](#)
 - `updateSecurityConfiguration(SecurityConfigDto)` (Java method), [386](#)
 - `updateSettings(String, String, Properties)` (Java method), [402](#), [406](#)
 - `updateSqlProperties()` (Java method), [79](#)
 - `updateUserDetailsWithoutPassword(UserDto)` (Java method), [388](#)
 - `updateUserDetailsWithPassword(UserDto)` (Java method), [388](#)
 - `UPLOAD_SIZE` (Java field), [83](#)
 - `URL_PATTERN` (Java field), [403](#)
 - `useFilter(Query, Filter)` (Java method), [242](#)
 - `useFilter(Query, List)` (Java method), [242](#)
 - `useFilter(Query, List, InstanceSecurityRestriction)` (Java method), [243](#)
 - `useFilter(Query, String[], Object[])` (Java method), [242](#)
 - `useFilter(Query, String[], Object[], InstanceSecurityRestriction)` (Java method), [242](#)
 - `useFilterFromPattern(Query, String, List)` (Java method), [243](#)
 - `USER_ACCESS_PREFIX` (Java field), [358](#)
 - `UserContextService` (Java interface), [389](#)
 - `UserDto` (Java class), [375](#)
 - `UserDto()` (Java constructor), [375](#)
 - `UserDto(MotechUser)` (Java constructor), [375](#)
 - `UserInfo` (Java class), [420](#)
 - `UserInfo(String, boolean, String)` (Java constructor), [420](#)
 - `USERNAME_PASSWORD` (Java field), [358](#)
 - `UsernameValueGenerator` (Java class), [236](#)
 - `UserNotFoundException` (Java class), [369](#)
 - `UserNotFoundException()` (Java constructor), [369](#)
 - `UserNotFoundException(String)` (Java constructor), [369](#)
 - `UserRegistrationValidator` (Java class), [427](#)
 - `UserRegistrationValidator(PersistedUserValidator, OpenIdUserValidator)` (Java constructor), [427](#)
 - `USERS` (Java field), [297](#)
 - `Util` (Java class), [287](#)
- ## V
- `validate(ResetForm)` (Java method), [426](#)
 - `validate(StartupForm, ConfigSource)` (Java method), [426](#)
 - `validate(StartupForm, List, ConfigSource)` (Java method), [423–425](#), [427](#)
 - `validateCredentials()` (Java method), [255](#)
 - `validateCredentials(T)` (Java method), [255](#)
 - `validateEmptyOrWhitespace(Errors, String, String)` (Java method), [427](#)
 - `validateNoJavaKeyword(String)` (Java method), [300](#)
 - `validateToken(String)` (Java method), [388](#)
 - `validateTokenAndLoginUser(String, HttpServletRequest, HttpServletResponse)` (Java method), [388](#)
 - `validateUrl(Errors, String)` (Java method), [427](#)
 - `ValidationCriterionDto` (Java class), [210](#)
 - `ValidationCriterionDto()` (Java constructor), [210](#)
 - `ValidationCriterionDto(String, TypeDto)` (Java constructor), [210](#)
 - `ValidationCriterionDto(String, TypeDto, Object, boolean)` (Java constructor), [210](#)
 - `ValidationException` (Java class), [480](#)
 - `ValidationException(String, Set)` (Java constructor), [480](#)
 - `ValidationUtil` (Java class), [300](#)
 - `ValidationUtils` (Java class), [427](#)
 - `VALUE` (Java field), [181](#), [199](#), [280](#)
 - `value()` (Java method), [335](#)
 - `valueAsString()` (Java method), [211](#)
 - `valueOf(String)` (Java method), [72](#), [89](#), [400](#)
 - `valuesForQuery()` (Java method), [222](#)
 - `VERSION` (Java field), [474](#)
 - `VIEW_BASIC_EMAIL_LOGS_PERMISSION` (Java field), [357](#)

VIEW_DETAILED_EMAIL_LOGS_PERMISSION

(Java field), [357](#)

VIEW_SECURITY (Java field), [357](#)

vote(Authentication, Object, Collection) (Java method),
[351](#)

W

waitForContext(int) (Java method), [54](#), [310](#)

WallTime (Java class), [78](#)

WallTime(String) (Java constructor), [78](#)

WallTimeUnit (Java enum), [79](#)

WARN (Java field), [49](#)

warn(String, String) (Java method), [52](#)

warn(String, String, DateTime) (Java method), [52](#)

WARNING (Java field), [466](#)

WebBundleUtil (Java class), [326](#)

WebUIBundleActivator (Java class), [318](#)

Wednesday (Java field), [70](#)

Week (Java field), [79](#)

WeeklyCronJobExpressionBuilder (Java class), [328](#)

WeeklyCronJobExpressionBuilder(DayOfWeek) (Java
constructor), [328](#)

WeeklyCronJobExpressionBuilder(int) (Java construc-
tor), [328](#)

WEEKS (Java field), [137](#)

with(String, String) (Java method), [66](#)

withAny(String, Set) (Java method), [66](#)

withDate(String, DateTime) (Java method), [66](#)

withDateRange(String, Range) (Java method), [66](#)

withDeliveryStatuses(DeliveryStatus) (Java method), [104](#)

withDeliveryStatuses(Set) (Java method), [104](#)

withDescription(String) (Java method), [467](#)

withField(String, String, String) (Java method), [66](#)

withFromAddress(String) (Java method), [104](#)

withId(Long) (Java method), [467](#)

withInt(String, int) (Java method), [67](#)

withMessage(String) (Java method), [105](#)

withMessageTime(DateTime) (Java method), [105](#)

withMessageTimeRange(Range) (Java method), [105](#)

withName(String) (Java method), [467](#)

withQueryParams(QueryParams) (Java method), [105](#)

withRepeatIntervalInDays(int) (Java method), [328](#)

withServiceName(String) (Java method), [484](#)

withSubject(String) (Java method), [106](#)

withTaskConfig(TaskConfig) (Java method), [467](#)

withTime(Time) (Java method), [328](#)

withToAddress(String) (Java method), [106](#)

withTrigger(TaskTriggerInformation) (Java method), [467](#)

WRITABLE (Java field), [88](#)

Y

YEARS (Java field), [137](#)

YES (Java field), [223](#)