

Working with Annotations

- [AOP Annotations](#)
 - [General information](#)
 - [Using annotations](#)
 - [Parameters](#)
 - [Annotations](#)
 - [Monitor](#)
 - [DontMonitor](#)
 - [Count](#)
 - [CountByParameter](#)
- [CDI Annotations](#)
- [SQL Annotations](#)

AOP Annotations

General information

You can find examples of annotations usage in the examples project in GitHub: <https://github.com/anotheria/moskito-examples/tree/master/aop>.

The annotations themselves are part of the moskito-aop module and are located in the package: *net.anotheria.moskito.aop.annotation*.

Using annotations

MoSKito annotations are used with aspectj. The easiest way to use them in your *maven* project is to add following to your pom.

```
<build>
  <plugins>
    <plugin>
      <groupId>org.codehaus.mojo</groupId>
      <artifactId>aspectj-maven-plugin</artifactId>
      <version>1.4</version>
      <configuration>
        <aspectLibraries>
          <aspectLibrary>
            <groupId>net.anotheria</groupId>
            <artifactId>moskito-aop</artifactId>
          </aspectLibrary>
        </aspectLibraries>
        <source>1.6</source>
        <target>1.6</target>
      </configuration>
      <executions>
        <execution>
          <goals>
            <goal>compile</goal>
          </goals>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>
...

```

of course you should add dependency to the moskito-annotation lib somewhere first.

```
<dependency>
  <groupId>net.anotheria</groupId>
  <artifactId>moskito-aop</artifactId>
  <version>${moskito.version}</version>
</dependency>

```

Alas, we recommend to add dependencies to **moskito-core**, **moskito-webui**, **moskito-annotations** explicitly and in same version.

Parameters

@Monitor, *@Count* and *@CountByParameter* as well as future monitoring enabling annotations support three optional parameters.

Parameter	Meaning	Default
producerId	The id of the producer	Class name
subsystem	Subsystem of the producer	"default"
category	Category of the producer	"annotated"

Annotations

Monitor

If used on a class - forces MoSKito to monitor the whole class including all methods, except those that are annotated with *DontMonitor*. Note that Monitor uses execution style pointcuts, therefore only the implementation code is monitored and has to be compile-time weaved.

DontMonitor

Forces MoSKito to ignore annotated method in a class annotated with *@Monitor*.

Count

Similar to *@Monitor* but instead of monitoring the target class/method as a Service a simple counter monitoring is applied.

CountByParameter

Can be applied to methods and uses the first parameter to the method as case separation.

CDI Annotations

TBD

SQL Annotations

TBD