

## ToStringGenerator

Since Log4E Pro 1.0 the Jakarta Commons StringGenerator is introduced. The StringGenerator is able to display nearly every datastructure. E.g.

```
logger.debug(new String[]{"foo"});
will print something like [Ljava.lang.String;@b166b5
to the output. With that approach, the values of the array are not readable. Using logger.debug(
    new ToStringBuilder("", MethodParameterStyle.METHOD_PARAMETER_STYLE).
    append(new String[]{"foo"}).toString());
will print {foo}
```

To use the StringGenerator, one has to install it for him or herself. It is included in the Jakarta Commons Lang Package available at <http://jakarta.apache.org/commons/lang/>

Download the latest distribution and include the commons-lang.jar in your CLASSPATH.

Log4E provides its own ToStringStyle called METHOD\_PARAMETER\_STYLE. One can define his or her own ToStringStyle or let create Log4E the predefined MethodParameterStyle in his or her own project. To let create the MethodParameterStyle source code by Log4E, just press the "Create ToStringStyle class ..." button. This has to be done once for each project.

Options:

- Use StringGenerator

If checked the Jakarta Commons StringGenerator is used to print the parameters.

Example generated code:

```
logger.debug("myMethod(" +
new ToStringBuilder("", MethodParameterStyle.METHOD_PARAMETER_STYLE).
append("String str", str).toString() + ") - start");
```

The METHOD\_PARAMETER\_STYLE will be discussed in the option "ToStringStyle"

Note that the ToStringBuilder is not used if no variable values has to be displayed. That means that even if "Use StringGenerator" is checked, Log4E will generate the following code if the preference "Log4E->Positions->Method Exit->Log with parameter values of method" is not checked: `logger.debug("myMethod(String str) - end");`

- StringGeneratorClass

Default: `org.apache.commons.lang.builder.ToStringBuilder`

- Dummy paramter (1st parameter of ToStringBuilder Constructor)

Unfortunately the ToStringBuilder throws an NullPointerException if the first constructor parameter is set to null. Thus, you have to set a dummy object which has no effect if using the MethodParameterStyle created by Log4E.

- ToStringStyle (2nd parameter of ToStringBuilder Constructor)

Default: `de.jayefem.log4e.MethodParameterStyle.METHOD_PARAMETER_STYLE`

The MethodParameterStyle has to be created in your project before using it. Use "Create ToStringStyle class ..." to create the source code in your project.

- Create ToStringStyle class ...

Using the button will pop up a dialog from which you can choose in which folder the source code of

MethodParameterStyle should be created. By default it will create the class

`src/de/jayefem/log4e/MethodParameterStyle.java` (the path depends on your package settings of the property

"ToStringStyle (2nd parameter of ToStringBuilder Constructor)".

It will generate the following code:

```
package de.jayefem.log4e;
import org.apache.commons.lang.builder.ToStringStyle;
/**
 * The method parameter style for logging with Log4E.
 *
 * @author Jan-Friedrich Mutter, http://log4e.jayefem.de
 */
public class MethodParameterStyle extends ToStringStyle {
/**
 * The method parameter style for logging with Log4E.
 * @see #readResolve()
 */
public static final ToStringStyle METHOD_PARAMETER_STYLE =
new MethodParameterStyle();
```

```
/**
 * Constructor.
 */
public MethodParameterStyle() {
    super();

    this.setUseClassName(false);
    this.setUseIdentityHashCode(false);
    this.setUseFieldNames(true);
    this.setContentStart("");
    this.setContentEnd("");
    this.setFieldNameValueSeparator("=");
}

/**
 * <p>Ensure <code>Singleton</code> after serialization.</p>
 * @return the singleton
 */
private Object readResolve() {
    return METHOD_PARAMETER_STYLE;
}
}
```