

```
/*
Pro Spring
By Rob Harrop
Jan Machacek
ISBN: 1-59059-461-4
Publisher: Apress
*/
```

```
////////////////////////////////////
////
//File: replacement.xml
<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN" "http://www.springframework.org/dt
d/spring-beans.dtd">
<beans>
  <bean id="methodReplacer" class="FormatMessageReplacer"/>
  <bean id="replacementTarget" class="ReplacementTarget">
    <replaced-method name="formatMessage" replacer="methodReplacer">
      <arg-type>String</arg-type>
    </replaced-method>
  </bean>
  <bean id="standardTarget" class="ReplacementTarget"/>
</beans>
```

```
////////////////////////////////////
////
public class ReplacementTarget {

    public String formatMessage(String msg) {
        return "<h1>" + msg + "</h1>";
    }

    public String formatMessage(Object msg) {
        return "<h1>" + msg + "</h1>";
    }

    public void foo() {

    }
}
```

```
////////////////////////////////////
////
import java.lang.reflect.Method;

import org.springframework.beans.factory.support.MethodReplacer;

public class FormatMessageReplacer implements MethodReplacer {

    public Object reimplement(Object target, Method method, Object[] args)
        throws Throwable {

        if (isFormatMessageMethod(method)) {

            String msg = (String) args[0];
```

```

        return "<h2>" + msg + "</h2>";
    } else {
        throw new IllegalArgumentException("Unable to reimplement method "
            + method.getName());
    }
}

private boolean isFormatMessageMethod(Method method) {
    // check correct number of params
    if (method.getParameterTypes().length != 1) {
        return false;
    }

    // check method name
    if (!("formatMessage".equals(method.getName()))) {
        return false;
    }

    // check return type
    if (method.getReturnType() != String.class) {
        return false;
    }

    // check parameter type is correct
    if (method.getParameterTypes()[0] != String.class) {
        return false;
    }

    return true;
}
}

```

```

////////////////////////////////////
////
import org.springframework.beans.factory.BeanFactory;
import org.springframework.beans.factory.xml.XmlBeanFactory;
import org.springframework.core.io.FileSystemResource;
import org.springframework.util.StopWatch;

public class MethodReplacementExample {

    public static void main(String[] args) {
        BeanFactory factory = new XmlBeanFactory(new FileSystemResource(
            "build/replacement.xml"));

        ReplacementTarget replacementTarget = (ReplacementTarget) factory
            .getBean("replacementTarget");
        ReplacementTarget standardTarget = (ReplacementTarget) factory
            .getBean("standardTarget");

        displayInfo(replacementTarget);
        displayInfo(standardTarget);
    }

    private static void displayInfo(ReplacementTarget target) {

```

```
System.out.println(target.formatMessage("Hello World!"));

StopWatch stopWatch = new StopWatch();
stopWatch.start("perfTest");

for (int x = 0; x < 1000000; x++) {
    String out = target.formatMessage("foo");
}

stopWatch.stop();

System.out.println("1000000 invocations took: "
    + stopWatch.getTotalTimeMillis() + " ms");
}
}
```