

JBoss Community



Java Microbenchmarking Harness

Motivation

“The URL Validator uses the *java.net.URL* constructor to decide if an URL is valid. This is costly in terms of performance, since the URL class checks for the available URLStreamHandlers. This (a regular expression) is a much better and faster strategy: ...”

— HV-920

```
public class User {  
    @URL  
    private String homepage;  
  
    //...  
}
```

How hard can it be!?

```
private static final int MAX_ELEMENTS = 10_000_000;

public static void main(String[] args) {
    List<? extends Set<Integer>> testees =
        Arrays.asList(
            new HashSet<Integer>(),
            new TreeSet<Integer>(),
            new ConcurrentSkipListSet<Integer>());
    for (Set<Integer> testee : testees) {
        doBenchmark(testee);
    }
}
```


How hard can it be!?

```
private static void doBenchmark(Set<Integer> testee) {  
    long start = System.currentTimeMillis();  
    for (int i = 0; i < MAX_ELEMENTS; i++) {  
        testee.add(i);  
    }  
    long end = System.currentTimeMillis();  
    System.out.printf("%s took %d ms%n",  
        testee.getClass(), (end - start));  
}
```

Problems

- Garbage collection
- JIT and On-Stack-Replacement (OSR)
- Dead code elimination
- System load

Enter JMH

- Developed by Oracle/Sun Performance Engineering team
- Lead by Aleksey Shipilëv
- Roots in JRockit
- JMH 0.1 was released to the unsuspecting public in March 2013
- Current version 1.3

Getting started

```
mvn archetype:generate \  
    -DinteractiveMode=false \  
    -DarchetypeGroupId=org.openjdk.jmh \  
    -DarchetypeArtifactId=jmh-simple-benchmark-archetype \  
    -DgroupId=org.sample \  
    -DartifactId=test \  
    -Dversion=1.0
```

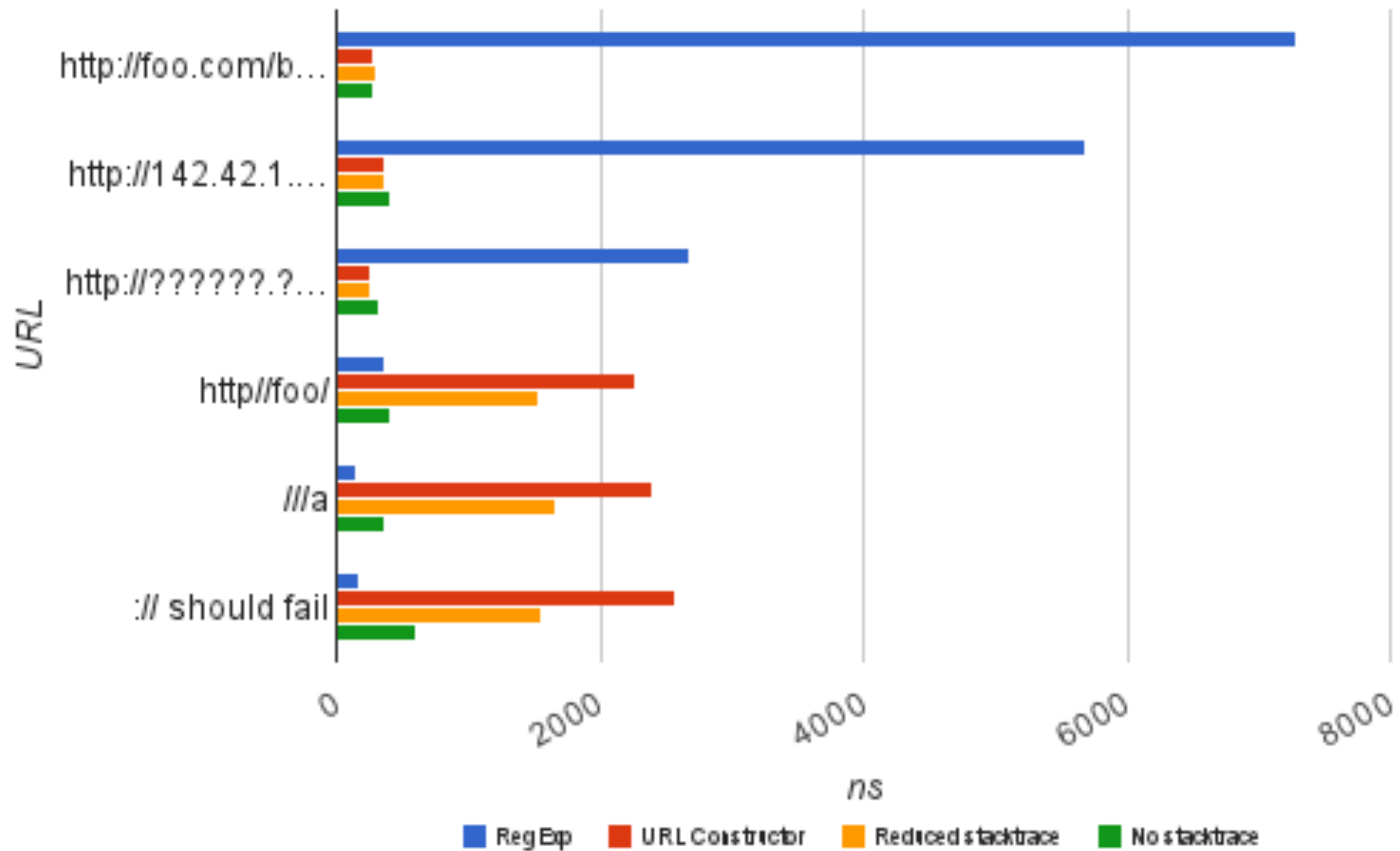
maven

Demo

Summary

- Annotation driven
- Annotation Processors (JSR 269) create benchmark classes
- Run via shaded jar file or as executing the Runner directly
- Configure via annotations, fluent API or command lines flags

URL constraint perf



Links

- <http://openjdk.java.net/projects/code-tools/jmh>
- <http://daniel.mitterdorfer.name/categories/series-jmh-intro>
- <http://psy-lob-saw.blogspot.se/p/jmh-related-posts.html>
- <http://stas-blogspot.blogspot.se/2011/07/most-complete-list-of-xx-options-for.htm>

<https://github.com/hferentschik/jmh-perf-tests>



user:hferentschik

BTW, it's a regexp monster!

[illegible]

Q + A