



PerfCake

Server applications' performance measurement



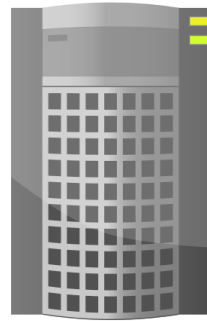
Pavel Macík (pmacik@perfcake.org)

Jiří Sedláček (jiri@sedlackovi.cz)

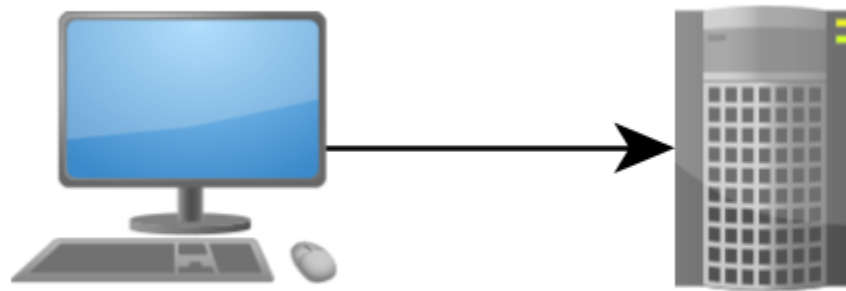
Martin Večeřa (mvecera@perfcake.org)

www.perfcake.org

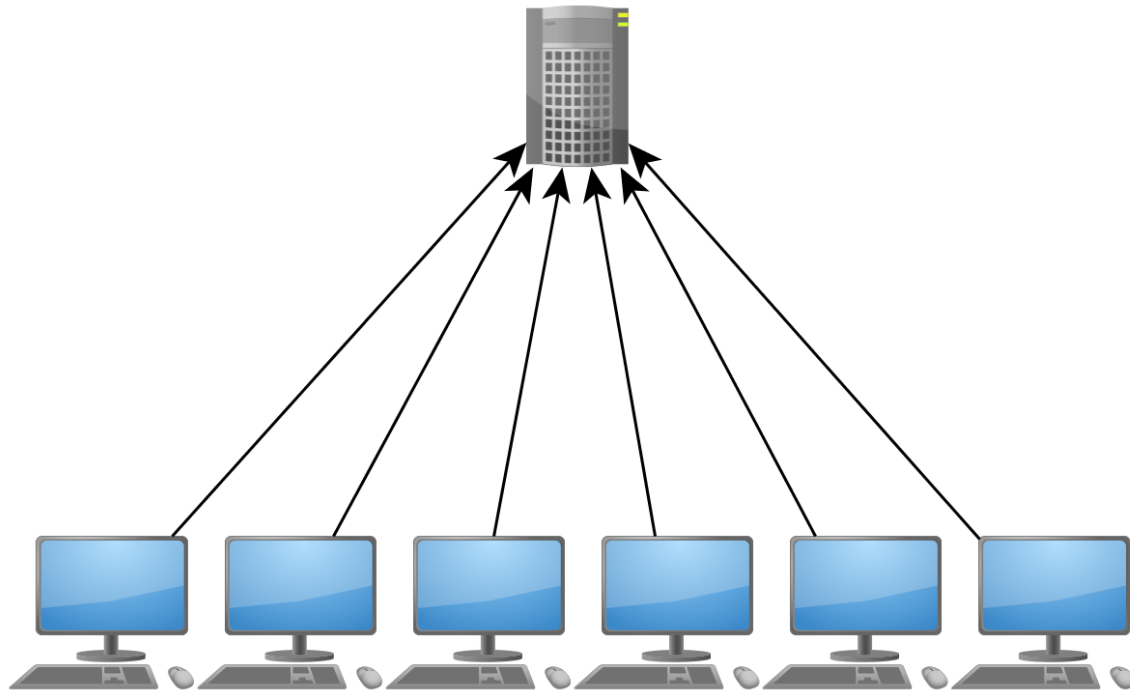
Performance Measurement



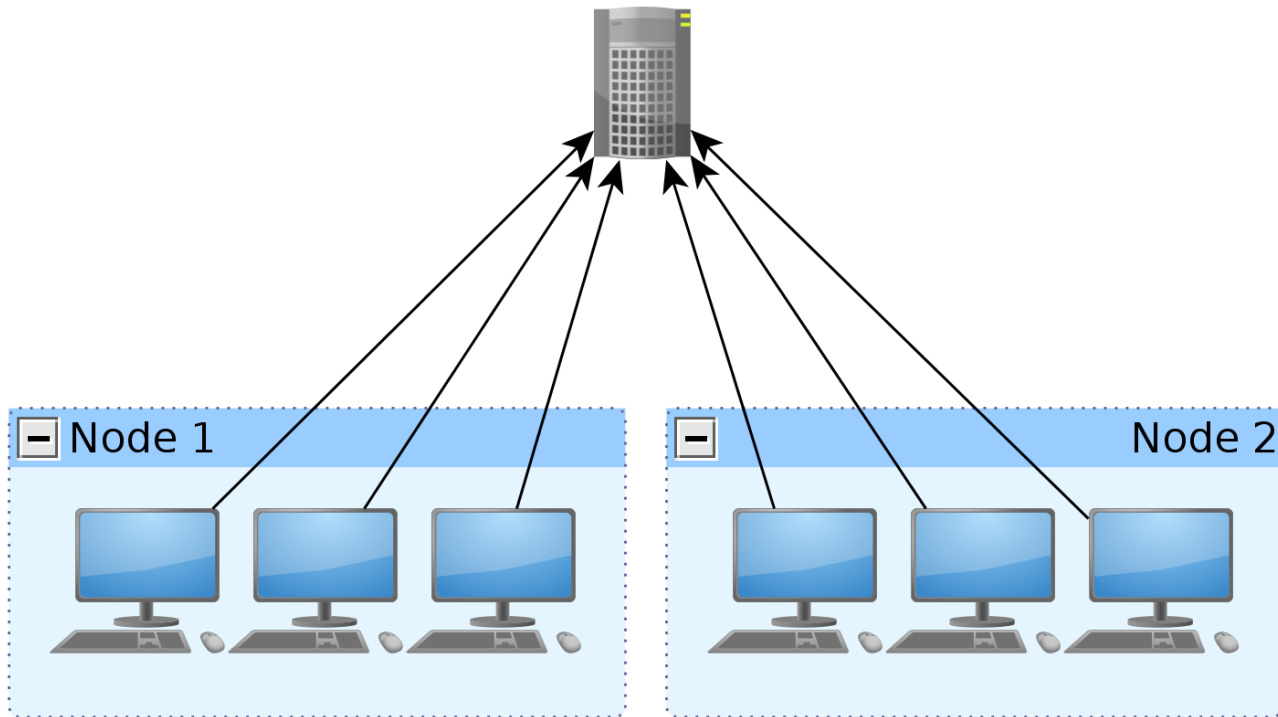
Performance Measurement



Performance Measurement



Performance Measurement



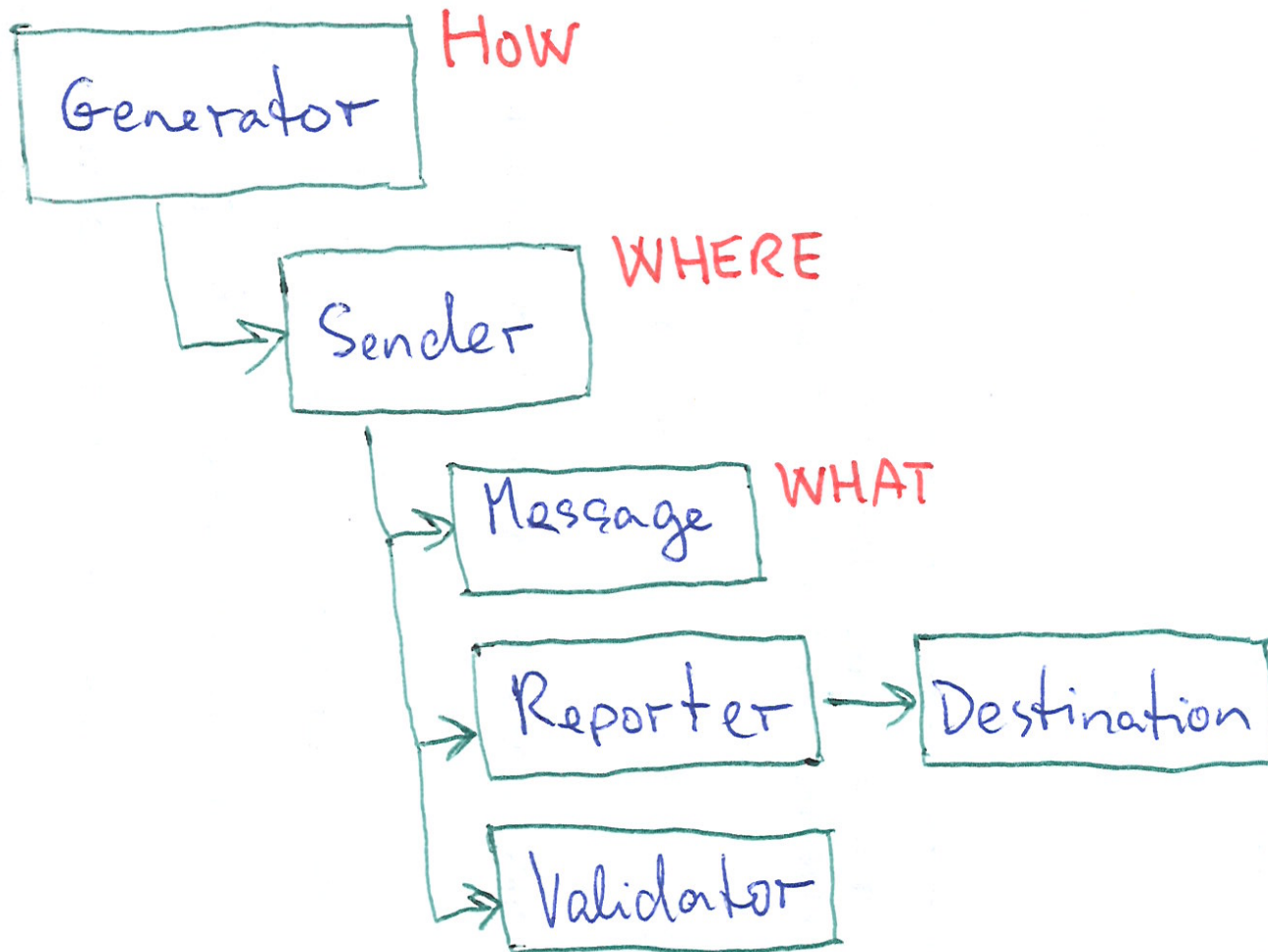
How to measure?

- number of clients/threads
- size of the request/message
- backlog, throughput, ramp up
- number of iterations, measurement time
- response time, memory consumption
- measurement frequency over time

Measurement Reliability

- application warm-up
- influence on the application
- measurement tool throughput
- reproducible results, small jitter
- valid response of the application
- stable environment

PerfCake Architecture



Configuration

```
<?xml version="1.0" encoding="utf-8"?>
<scenario xmlns="urn:perfcake:scenario:1.0">
  <generator class="DefaultMessageGenerator" threads="${thread.count:10}">
    <run type="time" value="10000"/>
    <property name="threadQueueSize" value="5000"/>
  </generator>
  <sender class="HTTPSender">
    <property name="target" value="http://your-app.com/cool-app"/>
  </sender>
  <reporting>
    <reporter class="ResponseTimeReporter">
      <destination class="ConsoleDestination">
        <period type="time" value="2000"/>
      </destination>
    </reporter>
  </reporting>
</scenario>
```

Extending PerfCake

```
38 public class DummySender extends AbstractSender {
40*   * The sender's logger.[]
42   private static final Logger log = Logger.getLogger(DummySender.class);
43
45*   * The delay duration to simulate a asynchronous waiting.[]
47   private long delay = 0;
48
50*   * (non-Javadoc)[]
54*   @Override
55   public void init() throws Exception {
56     if (log.isDebugEnabled()) {
57       log.debug("Initializing... " + target);
58     }
59     // nop
60   }
61
63*   * (non-Javadoc)[]
67*   @Override
68   public void close() {
69     if (log.isDebugEnabled()) {
70       log.debug("Closing..");
71     }
72     // nop
73   }
74
76*   * (non-Javadoc)[]
80*   @Override
81   public Serializable doSend(final Message message, final Map<String, String> properties, final MeasurementUnit mu) throws Exception {
82     if (log.isDebugEnabled()) {
83       log.debug("Sending to " + target + "...");
84     }
85     if (delay > 0) {
86       Thread.sleep(delay);
87     }
88     // nop
89     return (message == null) ? message : message.getPayload();
90   }
91
93*   * Used to read the value of delay.[]
97*   public long getDelay() {[]
100
102*   * Sets the value of delay.[]
107*   public void setDelay(final long delay) {[]
110
111 }
```

Future Development

- IDE
- more protocols
- documentation
- build the community ;-)

ⓘ Add support for running PerfCake in cluster enhancement

Opened by [marvec](#) 2 months ago

ⓘ Measure theoretical throughput enhancement

Opened by [marvec](#) 2 months ago


ⓘ Refactor validators enhancement

Opened by [marvec](#) 2 months ago

ⓘ Prepare binary distro with JRE enhancement

Opened by [marvec](#) 2 months ago

ⓘ Extend ScenarioExecution to load custom system properties from a proper file enhancement cake

Opened by [pmacik](#) 2 months ago  2 comments

ⓘ Update site to Bootstrap 3.0 enhancement cake

Opened by [marvec](#) 2 months ago

Links

- www.perfcake.org
- github.com/PerfCake/PerfCake/issues
- site.perfcake.org
- javadoc.perfcake.org
- perfcake.ci.cloudbees.com
- twitter.com/perfcaketool
- perfcake.blogspot.com



?

Demo!