CoreHttpDriver.java

```java
package com.testnscale.sampleweb;
import com.sun.faban.common.NameValuePair;
import com.sun.faban.common.Utilities;
import com.sun.faban.driver.*;
import com.sun.faban.driver.util.ContentSizeStats;
import javax.xml.xpath.XPathExpressionException;
import java.io.IOException;
import java.util.List;
import java.util.concurrent.TimeUnit;
/**
* Basic web workload driver drives only one operation via a get request. */
@BenchmarkDefinition (
name = "SampleWeb Workload", version = "0.1", metric = "req/s"
)
@BenchmarkDriver (
name = "SampleWeb", threadPerScale = (float)1, opsUnit = "requests", metric
= "req/s", responseTimeUnit = TimeUnit.MILLISECONDS
)
@FixedTime (
cycleType = CycleType.THINKTIME, cycleTime = 0, cycleDeviation = 5
)
@MatrixMix (
operations = {"HomePage", "ProductsPage", "ContactsPage"}, mix = { @Row ({
0, 80, 20 }), @Row ({20, 39, 41 }), @Row ({60, 19, 21 })
}
)
public class SampleWebDriver {
private DriverContext ctx;
private HttpTransport http;
private String url, homeUrl, productsUrl, contactsUrl;
ContentSizeStats contentStats = null;
/**
* Constructs the basic web workload driver. * @throws
XPathExpressionException An XPath error occurred
*/
public SampleWebDriver() throws XPathExpressionException,
ConfigurationException {
ctx = DriverContext.getContext();
HttpTransport.setProvider(
"com.sun.faban.driver.transport.hc3.ApacheHC3Transport");
http = HttpTransport.newInstance();
StringBuilder urlBuilder = new StringBuilder();
String s = ctx.getProperty("secure");
if ("true".equalsIgnoreCase(s))
urlBuilder.append("https://");
else
urlBuilder.append("http://");
s = ctx.getXPathValue("/sampleWeb/webServer/fa:hostConfig/fa:hostPorts");
```

```java
List<NameValuePair<Integer>> hostPorts = Utilities.parseHostPorts(s);
// We currently only support a single host/port with this workload
if (hostPorts.size() != 1)
throw new ConfigurationException(
"Only single host:port currently supported.");
NameValuePair<Integer> hostPort = hostPorts.get(0);
urlBuilder.append(hostPort.name);
if (hostPort.value != null)
urlBuilder.append(':').append(hostPort.value);
s = ctx.getProperty("contextPath");
if (s.charAt(0) == '/')
urlBuilder.append(s);
else
urlBuilder.append('/').append(s);
url = urlBuilder.toString();
s = ctx.getProperty("homePath");
if (s.charAt(0) == '/')
homeUrl = url + s;
else
homeUrl = url + '/' + s;
s = ctx.getProperty("productsPath");
if (s.charAt(0) == '/')
productsUrl = url + s;
else
productsUrl = url + '/' + s;
s = ctx.getProperty("contactsPath");
if (s.charAt(0) == '/')
contactsUrl = url + s;
else
contactsUrl = url + '/' + s;
contentStats = new ContentSizeStats(ctx.getOperationCount());
ctx.attachMetrics(contentStats);
}
/**
* Do the request. * @throws IOException An I/O or network error occurred. */
@BenchmarkOperation (
name = "Request", max90th = 250, // 250 millisec
timing = Timing.AUTO
)
public void doRequest() throws IOException {
int size = http.readURL(url);
if (ctx.isTxSteadyState())
contentStats.sumContentSize[ctx.getOperationId()] += size;
}
}
```