# SiteStats Testing

## 1. Setup

Load tests setup for SiteStats consists on the following:

### 1.1 Provisioning Sakai with sites, users and resources:

1. Download version_3 of the Provisioning scripts from Alan Berg
2. Download SiteStats Provisioning config for the Provisioning tool
3. Change to (2.) folder, create links for scripts from provisioning tool
4. Follow detailed instructions for remaining steps

### 1.2 Preparing load tests:

1. Download and configure The Grinder 3 and Jython
2. Download SiteStats Grinder config
3. If site property files from SiteStats Provisioning config were modified, sakai_users_on.txt and sakai_users_on.txt must be re-created (syntax: userId,userPwd,siteId,)
4. Adjust test parameters on grinder.properties file

### 1.3 Executing load tests:

The following steps are only a recommended sequence:

1. Start The Grinder console
2. Start The Grinder agents from (2.) folder, on any (other) machine
3. Log in as admin on destination Sakai server and reset SiteStats metrics by browsing to "http://SAKAI_HOST:SAKAI_PORT /direct/sitestats-metrics/reset-all-metrics"
4. Run The Grinder tests
5. Log in as admin on destination Sakai server and get SiteStats metrics by browsing to "http://SAKAI_HOST:SAKAI_PORT/direct /sitestats-metrics/get-all-metrics". This provides important SiteStats aggregation stats as "Average time spent in event processing per event", "Number of events processed per sec", "Number of events generated in Sakai per sec", etc.
6. Ask Sakai server admins for Apache logs

### 1.4 Processing results:

1. The Grinder: manually or by using Grinder Analyzer
2. Apache logs: apache-response-time or any other log analyzer
3. The SiteStats metrics (from step 5. above) can give an idea of the SiteStats real-time thread aggregator impact on the system. See section Other data below.

---

## 2. Results

Unknown macro: {composition-setup}

Unknown macro: {card}

### 1. Environment

- **Server:** qa1-nl.sakaiproject.org (tech specs)
- **Sakai version:** 2.7.0M2
- **SiteStats version:** 2.1.0-b01
- **Sites:**
  - 1) 50 sites with SiteStats
  - 2) 50 sites without SiteStats (won't process events)
- **Users:** 50 users per site
- ***The Grinder* configuration:**
  - 1 process, 50 threads, 50 runs
  - Thread distribution:
    - 50% threads: login, goto a site, download a file, logout (generate a resource event)
    - 25% threads: login, goto a site, logout (generate a site visit event)
    - 25% threads: login, goto a site, access news tool, logout (generate other tool event)

### 2. Results

### 2.1. Charts & Raw data

- **GrinderAnalyzer charts & tables:** with SiteStats ON | with SiteStats OFF
- **SiteStats metrics:** with SiteStats ON | with SiteStats OFF
- **Raw data:** svn

### 2.2. Summary

| Test | Test Pass Rate | Concurrent Active Users | Mean Response Time | Resp. Time Std. Dev. | Mean Time to First Byte | # Events Generated in Sakai/sec | # Events Processed by SST /sec | Avg. Time to Process Event |
|---|---|---|---|---|---|---|---|---|
| Test (1) sites (SiteStats enabled) | 1.0 | 40-50 (1) | 1065.19 ms (-4.5%) | 1846.16 ms | 1032.63 ms | 12.639 | 268.48 | 3 ms |
| Test (2) sites (SiteStats disabled) | 1.0 | 40-50 (1) | 1113.15 ms | 2477.04 ms | 1078.85 ms | 13.327 | n.a. | 0 ms |

Notes

(1): This number doesn't reflect typical production numbers of concurrent users or active users - these are highly active users generating activity in milliseconds interval (in a real production system, this could be equivalent to more than 200 concurrent/active users).

Unknown macro: {card}

## 1. Environment

- **Server:** qa1-nl.sakaiproject.org (tech specs)
- **Sakai version:** 2.7.0M2
- **SiteStats version:** 2.1.0-b01
- **Sites:**
  - 1) 10 sites with SiteStats
  - 2) 10 sites without SiteStats (won't process events)
- **Users:** 500 users per site
- ***The Grinder* configuration:**
  - 1 process, 200 threads, 25 runs
  - Thread distribution:
    - 50% threads: login, goto a site, download a file, logout (generate a resource event)
    - 25% threads: login, goto a site, logout (generate a site visit event)
    - 25% threads: login, goto a site, access news tool, logout (generate other tool event)

## 2. Results

### 2.1. Charts & Raw data

- **GrinderAnalyzer charts & tables:** with SiteStats ON | with SiteStats OFF
- **SiteStats metrics:** with SiteStats ON | with SiteStats OFF
- **Raw data:** svn

### 2.1. Summary

| Test | Test Pass Rate | Concurrent Active Users | Mean Response Time | Resp. Time Std. Dev. | Mean Time to First Byte | # Events Generated in Sakai/sec | # Events Processed by SST /sec | Avg. Time to Process Event |
|---|---|---|---|---|---|---|---|---|
| Test (1) sites (SiteStats enabled) | 0.981 (1) | 150-200 (2) | 4901.9 ms (+2.17%) | 3596.49 ms | 4764.43 ms | 34.534 | 134.231 | 7 ms |
| Test (2) sites (SiteStats disabled) | 0.856 (1) | 150-200 (2) | 4797.62 ms | 3461.35 ms | 4683.58 ms | 31.357 | n.a. | 0 ms |

Notes

(1): These tests generated so stress that the server ran out of threads available - server responded with *503: Service Temporarily Unavailable*. Also, 2nd test (with SiteStats Off) completed with an higher rate of errors, affecting the results (it's faster to process a 503 page than serving processed page from Sakai).
(2): This number doesn't reflect typical production numbers of concurrent users or active users - these are highly active users generating activity in milliseconds interval (in a real production system, this could be equivalent to more than 500 concurrent/active users).

Unknown macro: {deck}

# 3. Other data

The SiteStats metrics (http://SAKAI_HOST:SAKAI_PORT/direct/sitestats-metrics/get-all-metrics), available on SiteStats 2.1+, can give an idea of how the SiteStats real-time thread aggregator is performing on a live system.

At UFP, we have a total throughput of 28.4 events processed/sec per server node for a rate of 0.15 Sakai events generated/sec per server node (full day stats). For day-only stats (when there is more traffic), we have a throughput of 14.28 events processed/sec per server node for a rate of 0.25 Sakai events generated/sec per server node. These values vary with Sakai load and DB load - since event processing interacts with DB, DB size and tunning are important. Our SST_EVENTS table currently have ~4.000.000 entries (as a reference, SAKAI_EVENT has 40.000.000 entries).