



sitestress.eu

load and performance testing



Einführung: Lasttests mit JMeter



Agenda

- Über SITESTRESS.EU
- Warum Lasttests?
- Was ist JMeter?
- Wie arbeitet JMeter?
- Skripte für JMeter
- Tests planen
- Testen
- Ergebnisse analysieren
- Beispiel JMeter-GUI





... über SITESTRESS.EU

- Webseiten schneller machen ist unser Thema!
- Spezialisiert auf Performance- und Lasttests und Optimierungen beim Client (on-PageSpeed)
- Seit 2003 in diesem Bereich unterwegs
- Tool-unabhängig (NeoLoad, JMeter, LoadRunner, Selenium)
- Agiles Testen
- Automatisiertes Testen
- Internet-Agentur-Background



Warum Lasttests?

- Ist-Aufnahme eines Systems
- Tuning nur mit reproduzierbaren Messergebnissen
- Durch Monitoring Flaschenhalse identifizieren
- Faster = better
- „Amazon noticed a 1% drop in sales for every 0.1 second the site loaded slower“
→ **Time = Money**





Was ist JMeter?

- JMeter erzeugt u.a. HTTP- und HTTPS-Zugriffe und somit Last auf beliebige Server
 - Kann mit sehr vielen verschiedenen Protokollen umgehen (SOAP, XML-RPC, http(s), JDBC)
 - Gute GUI für „Nicht-Programmierer“ aber auch Skriptfähig (JavaScript, Java)
 - Master-Slave / Client-Server-fähig, läuft in der Cloud → AmazonCloud EC2
- Open-Source-Software
 - frei verfügbar
 - Div. Plugins, erweiterbar
 - JMeter ist eine Java-Anwendung, läuft überall dort, wo auch Java läuft
- JMeter ermöglicht die Messung von Zugriffs- und Antwortzeiten von
 - Webanwendungen
 - Simuliert unterschiedliche virtuelle Benutzer (VU = Virtual User)



Wie arbeitet JMeter?

- Erstellung eines Skripts durch Aufzeichnung der Browserinteraktion mit einem Proxy (https wird seit JMeter 2.4 unterstützt)
- Parametrisierung des Skripts in der GUI (z.B.: Benutzernamen + Passwörter aus CSV-Liste, Servernamen, Port und Pfad aus Variablen)
- Test ausführen und Ergebnisse analysieren
- Jeder virtuelle User (VU) hat einen eigenen Thread
 - mehrere hundert VUs pro JMeter-Instanz
 - für große Lasten → mehrere Rechner → Amazon EC2



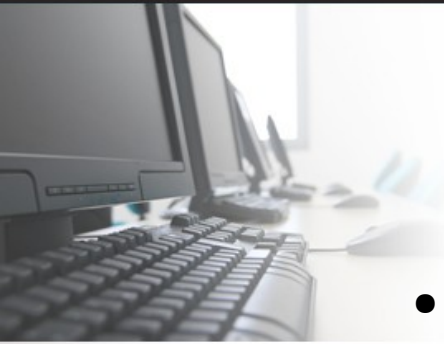
Skripte für JMeter

- Testfälle können mit einem „Rekorder“ aufgezeichnet werden
 - als Proxy realisiert
 - kann auch HTTPS aufzeichnen (Problem: Zertifikate → Man in the Middle)
 - Automatische Aufzeichnung von Cookies
 - Unterschiedliche Benutzer (VUs) mit unterschiedlichen Cookies realisierbar
- Zeitmessung für jeden Request und jeden VU → Achtung: große Datenmengen!
- Daten die per GET/POST/COOKIE übertragen werden, können parametrisiert werden, z.B. Benutzer/Passwort
- Für einfache bis mittlere Anforderung: Keine Programmierung oder Programmierkenntnisse notwendig, Einbindung von JavaScript, Java oder externe Programme in JMeter-Skripte für komplexe Szenarien möglich
- Sehr flexibles Variablenmanagement möglich
 - unterschiedliche „Variable Scopes“
 - Zufällig ausgewählte Listen oder der Reihenfolge nach



JMeter-Reports

- Div. Listener (von Aggregate-Reports bis CSV)
- Über Plugins leistungsfähiges Monitoring der Systeme machbar
- Das Datamining sollte in Excel oder R erfolgen
- Automatisierte Reporterstellung möglich (SLAs)





Tests planen

- Was soll genau getestet werden? (Loginvorgang, Seitenaufruf, Bestellvorgang)
- Eine oder mehrere Aufgaben/Skripte pro Test
- Nur ein Collector pro Test möglich
- Start und Ende definieren
- Anzahl an (unterschiedlichen) Benutzern definieren
- Wie werden die unterschiedlichen Benutzer aktiviert? (Gleichzeitig/Verzögert)



Testen

- Test vom Master aus starten
 - Slaves lassen sich auch einzeln starten
 - Übertragung der in den Slaves erhobenen Daten ist konfigurierbar (Batch, onHold, Testende, kontinuierlich)
 - Günstig: Slaves in der Cloud → Amazon EC 2
 - Kommunikation Master ↔ Slaves durch RMI
- Test anhalten durch
 - definiertes (Abbruch-)Kriterium
 - Benutzerinteraktion/Abbruch durch den Benutzer



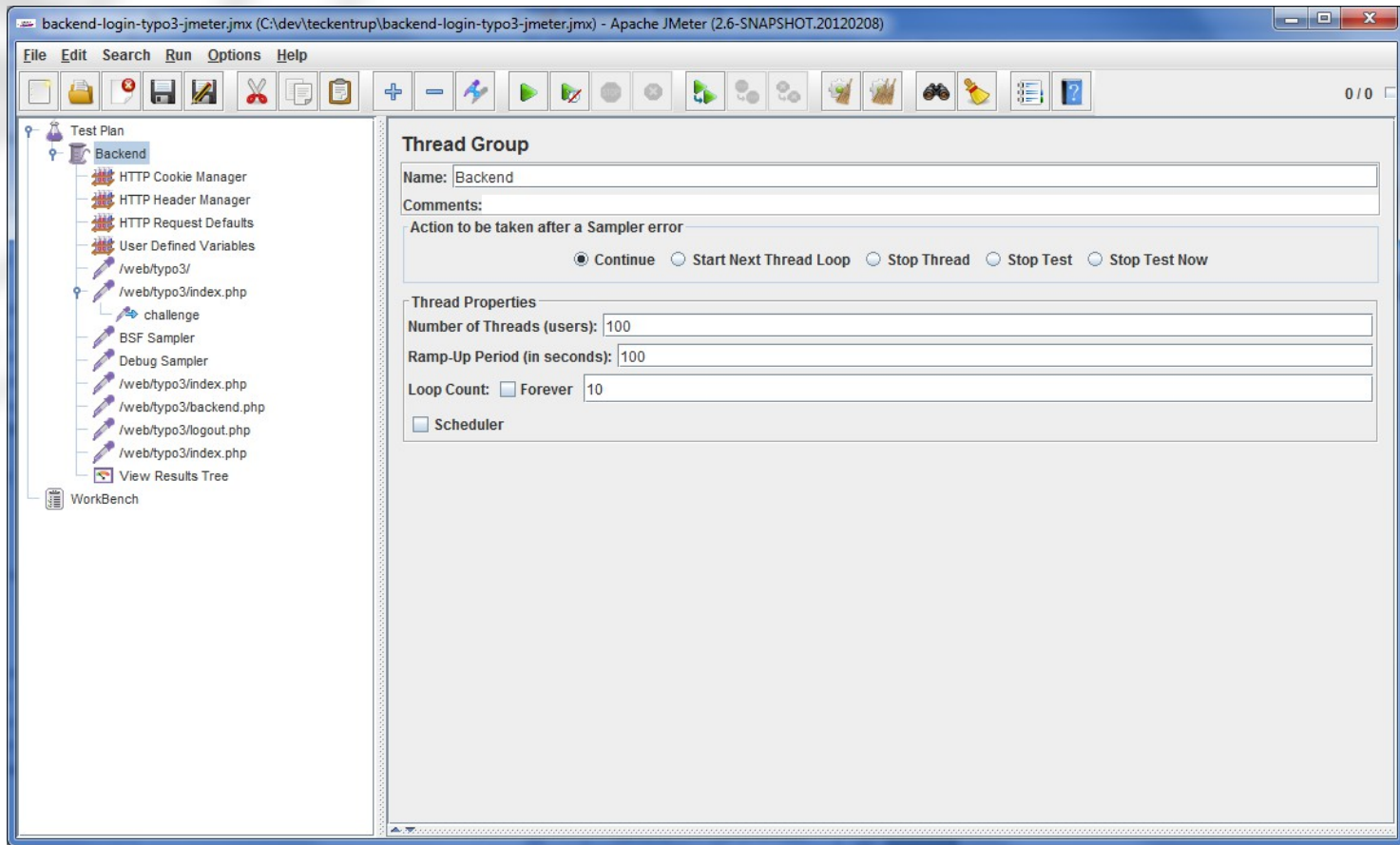
Ergebnisse analysieren

- Timer Values/Elapsed Time
- Timer Values/Anzahl an Benutzer (VUs)
- HTTP(S) Bytes/Sec
- HTTP Response Time vs. VUs
- Eigene Variablen können mitgeschrieben werden (z.B. Session-ID, IP des Clients, JavaScript-Hash)
- Export der Ergebnisse nach Excel oder spezielle Werkzeuge





Beispiel: JMeter-GUI





Danke für Ihre Aufmerksamkeit!

- Sitestress.eu ist ein Projekt der OGVIT GmbH & Co. KG
<http://www.sitestress.eu>

Tel.: +49 5251 687060

E-Mail: karl@sitestress.eu

- Paderborn:

- OGVIT GmbH & Co.KG
Jesuitenmuer 24
33098 Paderborn

- Hamburg:

- C/O Postalo GmbH
Kattrepel 2 - Montanhof
20095 Hamburg