



JMETER 정확히 알고 제대로 활용하자

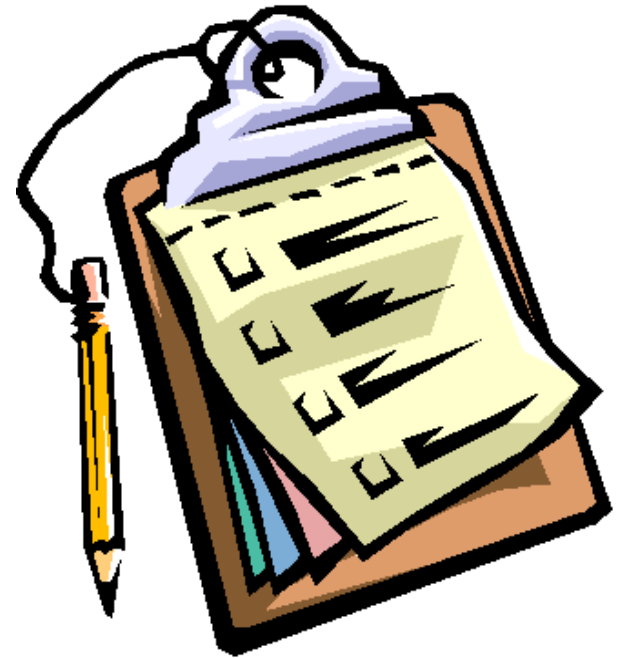
2015.04.29

STA 테스트 컨설팅



목차

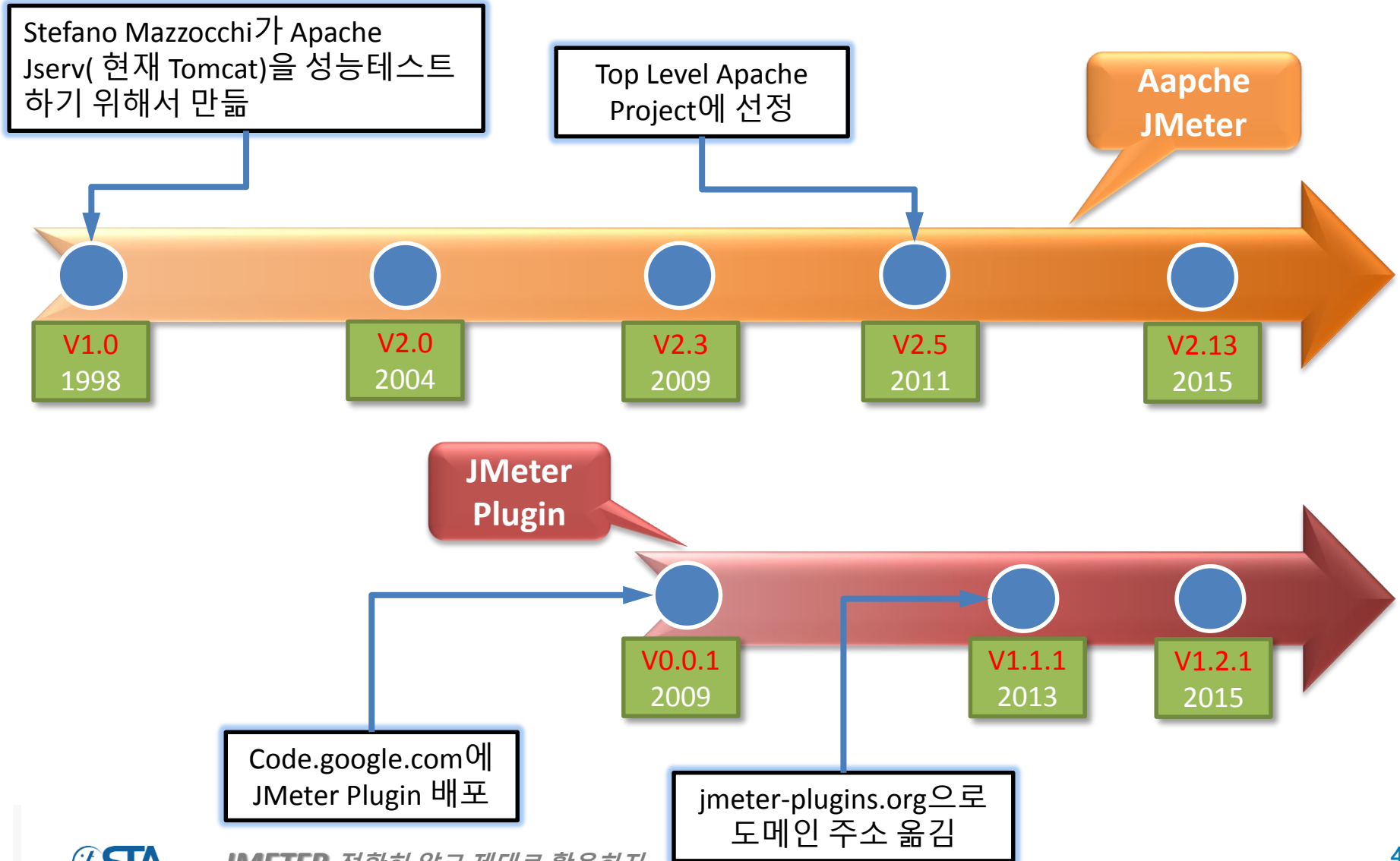
- JMeter 개요
- JMeter Plugin 소개
- Jmeter 주요 기능
 - 스크립트 작성
 - 부하발생 환경 구성
 - 서버 모니터링 설정
 - Workload 설계
 - 결과 데이터 저장 방법
 - 내장함수 생성
- 마무리



JMeter 개요

- Apache 재단의 Open Source
- 다양한 프로토콜 지원
 - Web 환경 : HTTP, HTTPS
 - Webservice 환경 : SOAP, REST
 - TCP 환경 : TCP
 - DB 환경 : JDBC
 - Big Data 환경 : MongoDB, Hadoop
 - 기타 : FTP, LDAP, JMS, SMTP, POP3, IMAP
- Plugin 구조

JMeter 발전 과정



JMeter 특징

- 사용자 수 무제한
 - License 수량 제한 없음
- 다양한 프로토콜 지원 및 지속적인 추가
 - MongoDB, Hadoop 추가
 - WebSocket은 곧 반영 예정
- 서버 Resource 모니터링
 - Unix, Windows, Mac에 대해서 75가지 성능지표 모니터링 가능
 - CPU, Memory, Disk I/O, NW I/O 등

JMeter 특징

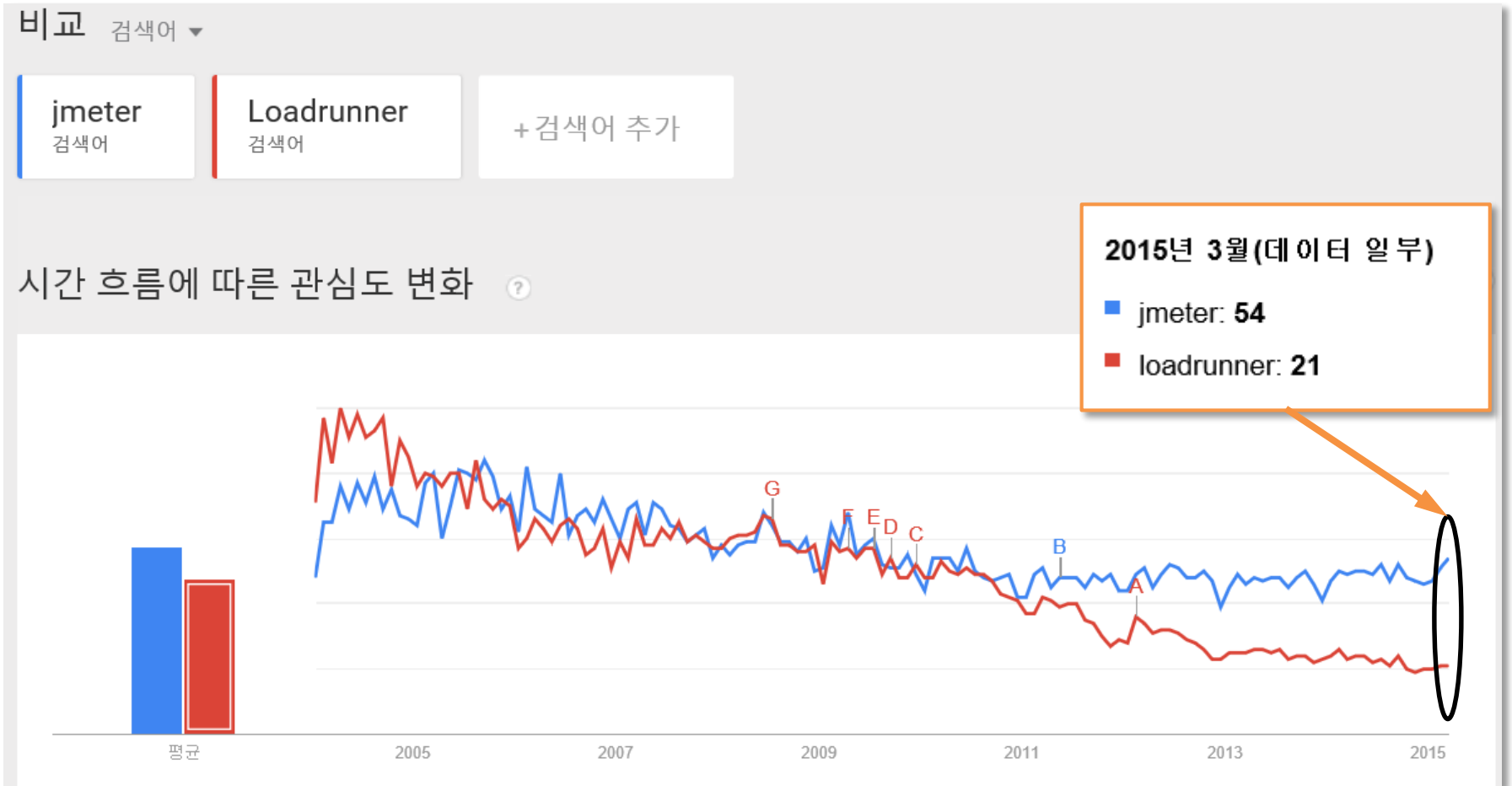
- 다양한 Plugin 제공
 - 국외 : <http://jmeter-plugins.org/>
 - 국내 : http://www.sten.or.kr/club/club_main.php?cb_id=cb_Jmeter
- Functional 테스트, Regression 테스트, 테스트 자동화 가능
 - 스크립트를 작성하고, Jenkins와 연동
- Light Weight Tool
 - 100% 자바로 구현
 - MAC, Unix/Linux, Windows에서 동작

JMeter 특징

- 여러 회사에서 유료 성능테스트 서비스 제공
 - [Blazemeter Ltd](#)
 - [Flood.IO](#)
 - [AppVance](#)
 - [LoadFocus](#)
 - [Loadosophia](#)
 - [JAnalyser](#)

.....

Google Trend - JMeter VS Loadrunner



JMeter Plugin 소개

- 독립적인 도메인
 - <http://jmeter-plugins.org>
- 5개의 카테고리로 50여가지 Plugin 제공

Standard Set

Basic plugins for everyday needs. Does not require additional libs to run.

[Download](#) | [Installation](#) | [Package Contents](#)



Extras Set

Additional plugins for extended and complex testing. Does not require additional libs to run.

[Download](#) | [Installation](#) | [Package Contents](#)



Extras with Libs Set

Additional plugins that *do require* additional libs to run.

[Download](#) | [Installation](#) | [Package Contents](#)



WebDriver Set

Selenium/WebDriver testing ability.

[Download](#) | [Installation](#) | [Package Contents](#)



Hadoop Set

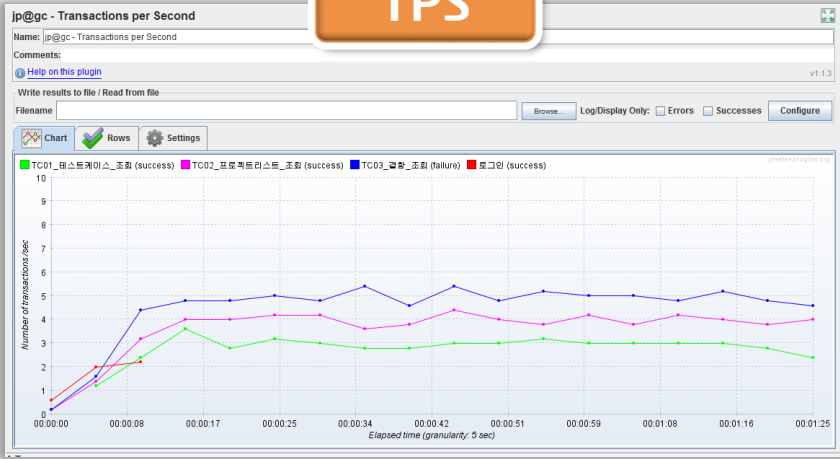
Hadoop/HBase testing plugins.

[Download](#) | [Installation](#) | [Package Contents](#)

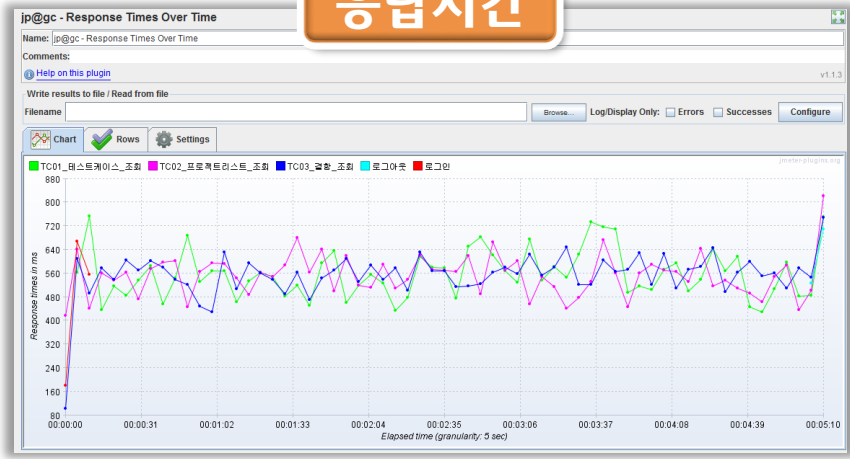


주요 JMeter Graph Plugin

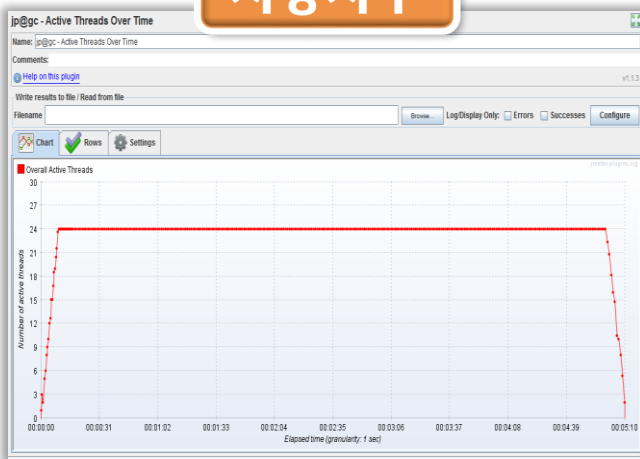
TPS



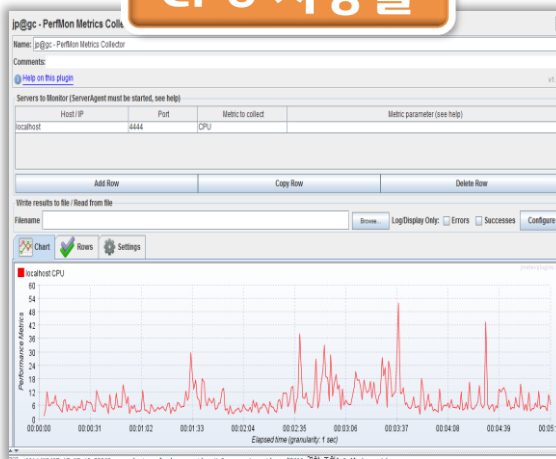
응답시간



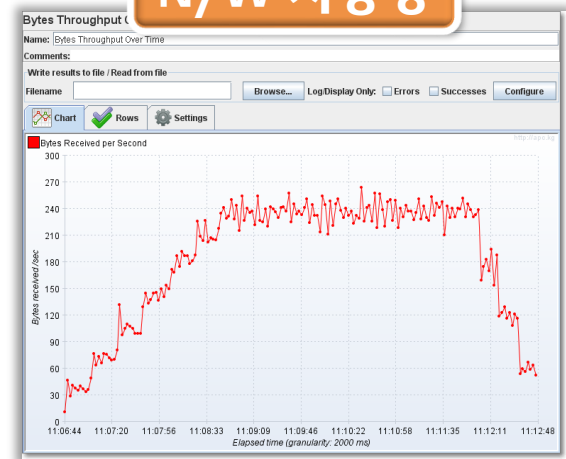
사용자수



CPU 사용률



N/W 사용량



주요 JMeter Graph Plugin

- WebDriver Set
 - Selenium을 이용하여 GUI 테스트 가능
 - Android
 - Chrome
 - Firefox

JMeter 이용 사례

Company	Country	Comment
ads-COM	France	We use JMeter for performance and fonctionnal testing.
ALALOOP	France	We use JMeter since 2008 for performance testing of many Web applications.
AOL	Worldwide	We use JMeter for performance testing, QA , and adhoc production requests.
Applabs Technologies Pvt.Ltd	India	We use JMeter for performance testing. We used Jmeter for more than 10 projects.
ARDAS group	Ukraine	We use JMeter since 2006 year for performance testing in all our java projects.
ARIS B.V.	Netherlands	We use JMeter for performance testing on the OSIRIS study tracking system and the Dutch Risc database and map for accidents and catastrophes.
AtlantBH	Bosnia-Herzegovina	We use JMeter for functional testing (e.g. web services), regression testing and data quality testing. We also use it for testing HBase/Hadoop .
Cofman	Denmark	We are using JMeter for load testing on our systems
Consology (Pty) Ltd	South Africa	E-billing
Corporate Express Australia Ltd.	Australia and NZ	Integration, load and stress testing B2B and EAI integration systems.
CSIRO	Australia	IM & T is using JMeter to test our managed applications and infrastructure.
DATACOM	New Zealand	Using JMeter to performance test our developed code. Most of it for local government agencies.
DataResolve Systems	India	We use JMeter for performance testing of our servers.
eBuilder Technology Centre	Sri Lanka	We use JMeter to perform load, functional & regression testing on the applications of e-TravelManagement, e-Procurement & e-Logistics business areas.
eFusion	Jakarta / Singapore	use JMeter in our performance & load testing of our applications.
Ephibian	Tucson, AZ	We develop enterprise web applications using open source software such as Tomcat, Apache, JMeter and PostgreSQL.

JMeterUsers (last edited 2011-09-27 15:16:39 by [Philippe Mouawad](#))

JMeter 이용 사례

Company	Country	Comment
kippdata informationstechnologie GmbH	Germany	We offer professional performance testing and trainings using Apache JMeter
Lufthansa Systems Indonesia	Indonesia	We use JMeter for performance testing. Currently we use it for testing our Frequent Flyer Program application.
Neev Information Technologies	India	We are using JMeter for Performance and Load testing of Web Applications we develop for our clients.
orbitz	USA	We use JMeter for performance testing in our DEV & DEVQA servers.
Panter	Switzerland	Performance regression testing of rails apps
Pragmatic Test Labs	Sri Lanka	We use JMeter for web applications performance testing. We provide JMeter trainings too.
PrepMe	USA	We wrap JMeter with a few scripts to dynamically generate configuration files based on the type of tests we want to run. This is integrated with our internal Symfony framework installation.
SharpMind	Germany	we use JMeter for load, functional and regression testing of client-server applications
Sourcepole	Switzerland	Benchmarking WMS
Talis	England	We are using it to performance test our new Talis Engage system.
UBIK-INGENIERIE	France	We use JMeter since 2007 for performance and load testing of various Web Applications (Intranet/Internet) including ECommerce websites (Number of concurrent users can vary from 1000 to 400000 per day) . We used it for testing various server side technologies including .NET, PHP, ASP, J2EE and Custom Java Servers. We have also built an in-house solution for reporting and plugins to ease scripting of GWT technology and improve performances of distributed testing.
University of Western Cape	South Africa	We are using JMeter to test our in-house written E-Learning application.
WatchMouse	Netherlands	Using jMeter to enable our customers to monitor transactions on their web site.
Sapient	Worldwide	Uses JMeter for performance and load testing for various applications.
xrow GmbH	Germany	Uses JMeter for performance and load testing of eZ Publish applications.
Infospectrum India Pvt. Ltd	India	Uses JMeter for performance and load testing.

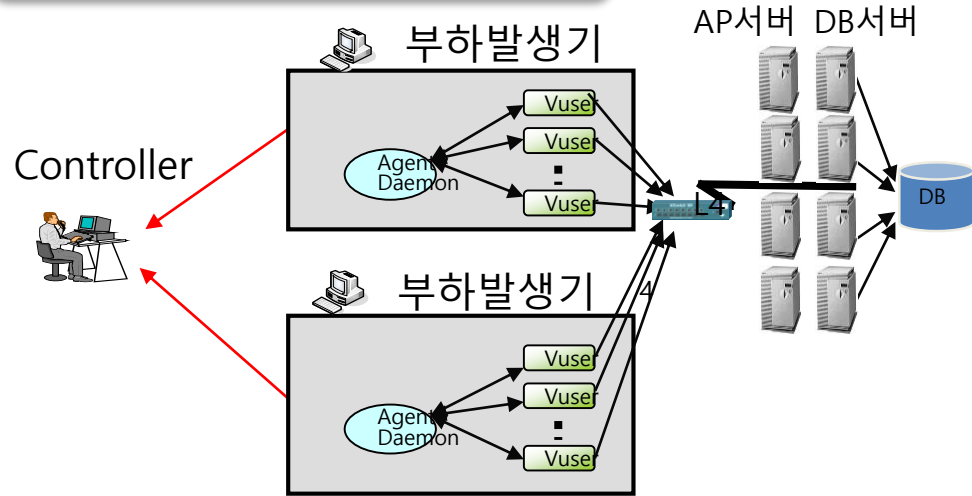
JMeterUsers (last edited 2011-09-27 15:16:39 by [Philippe Mouawad](#))

JMeter 구성 요소

Script 작성

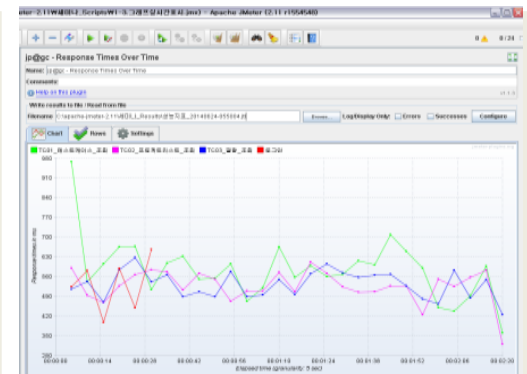
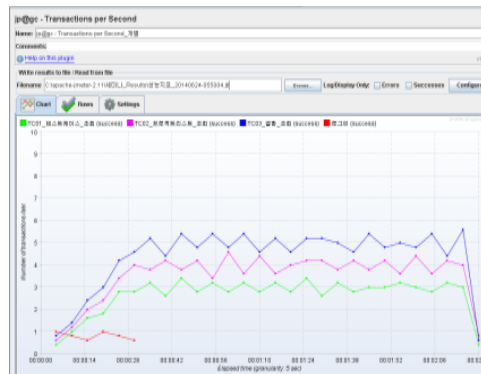
1. Recording
2. 파라미터 처리
3. Correlation 처리
4. Transaction 처리
5. Check Point 처리

컨트롤러와 부하발생기



결과 분석

1. Merge 기능
2. Filter 기능
3. Export 기능



스크립트 기본 구조

- Init 구간

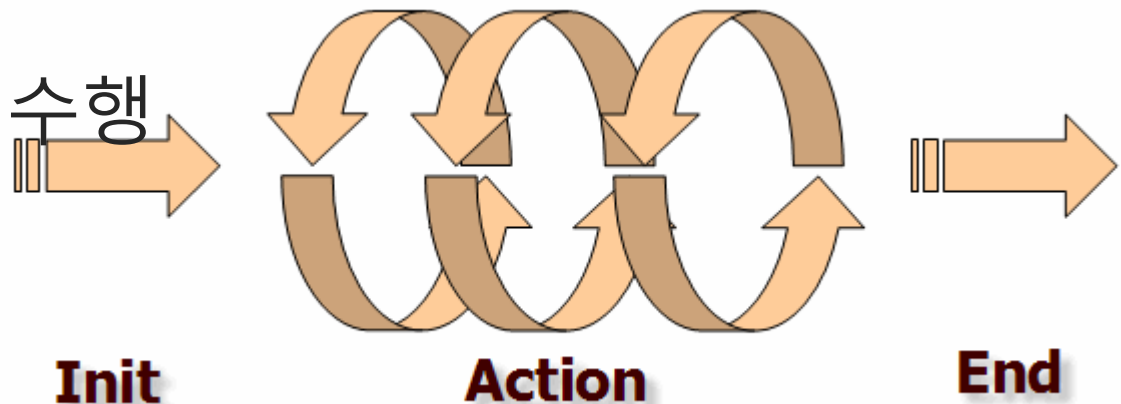
- 처음 한번만 수행 (ex. 로그인)

- 반복 구간

- 반복하고자 하는 트랜잭션을 적용(ex. 메일 조회)

- End 구간

- 마지막 한번만 수행 (ex. 로그아웃)



JMeter의 기본 스크립트 구조 적용

- Init 구간
 - Once Only Controller 이용
- 반복구간
 - 반복 회수 : Loop Controller 이용
 - 시간 : Runtime Controller 이용
- End 구간
 - Once Only Controller 이용

JMeter의 기본 스크립트 구조 적용

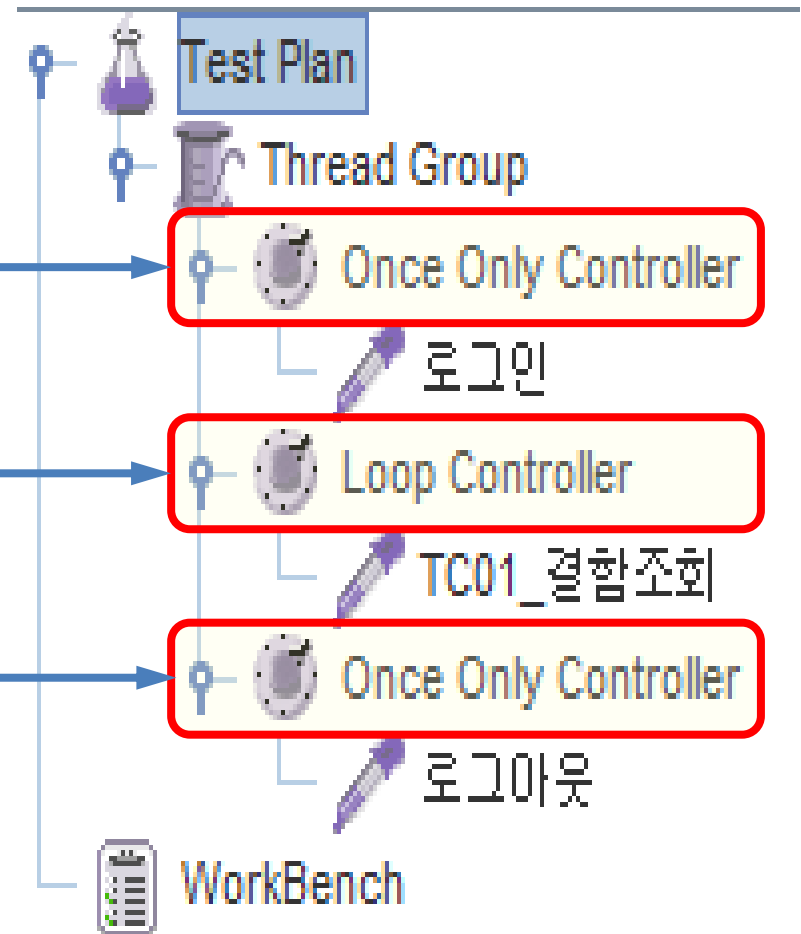
- Init 구간
 - Once Only Controller
- Loop에 의한 제어
- End 구간
 - Once Only Controller

Loop Controller

Name:

Comments:

Loop Count: Forever



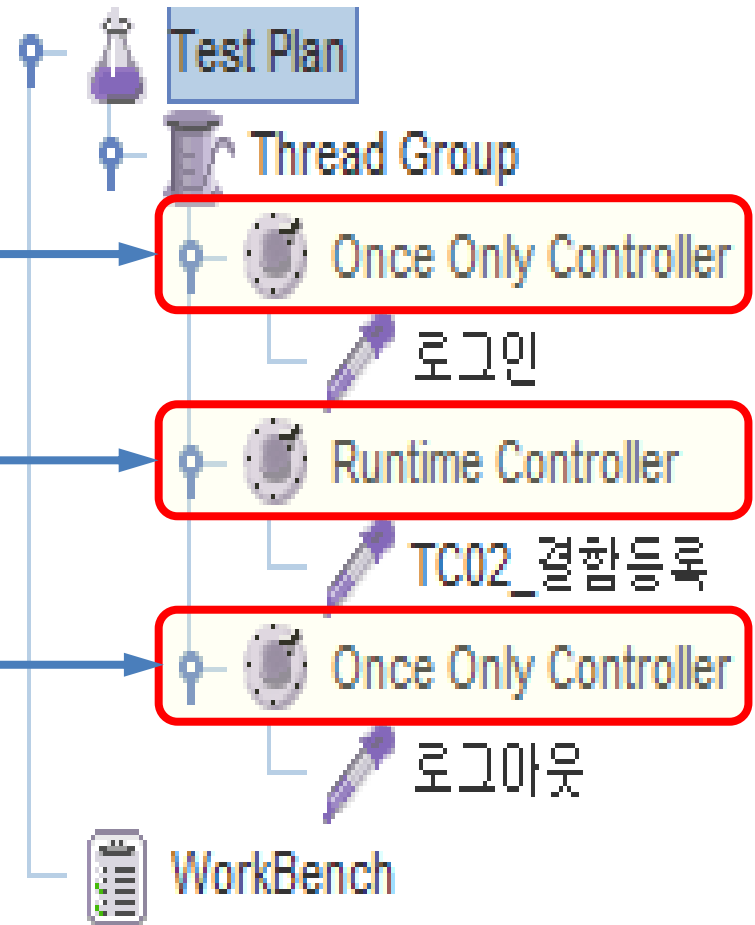
JMeter의 기본 스크립트 구조 적용

- Init 구간
 - Once Only Controller

- 시간에 의한 제어

Runtime Controller	
Name:	Runtime Controller
Comments:	
Runtime (seconds)	300

- End 구간
 - Once Only Controller



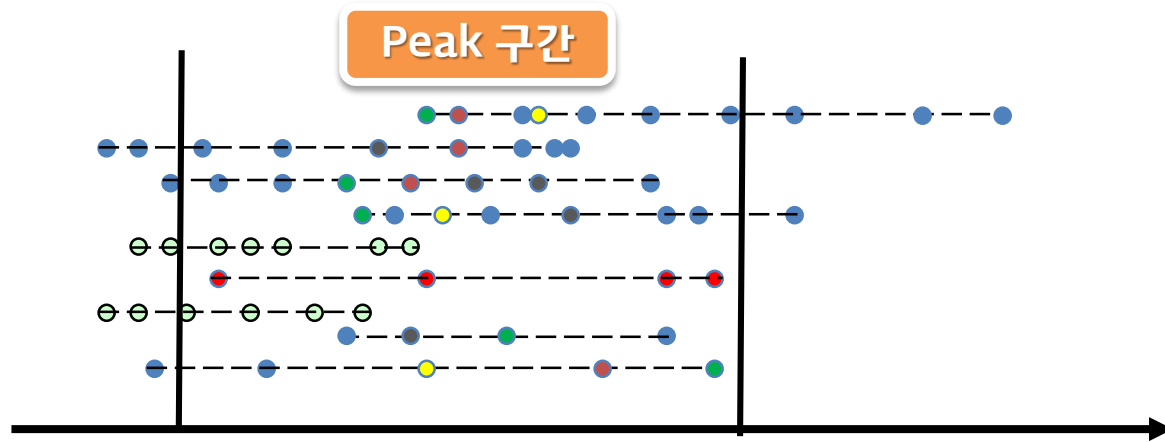
생각해보기 - 스크립트 작성

■ 메일 업무



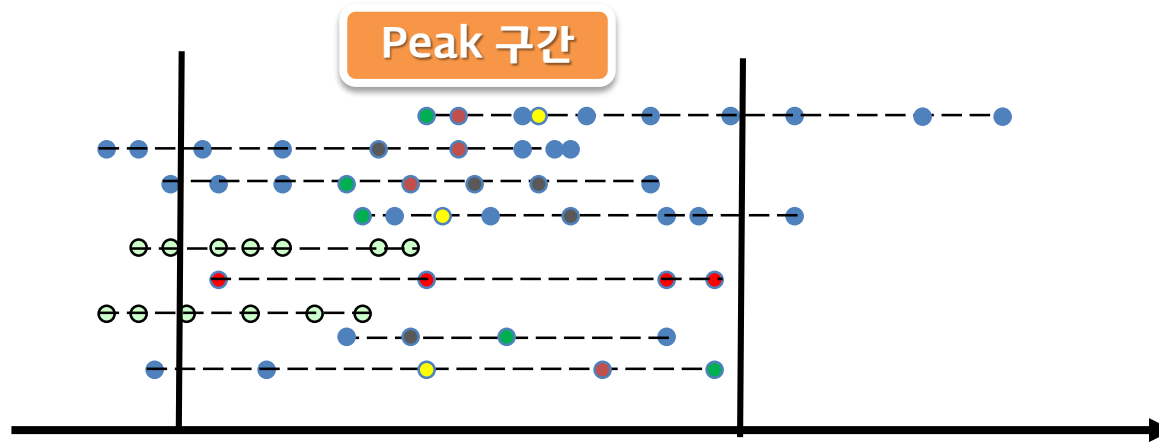
**어떤 방법으로
스크립트를 작성하고 계십니까?**

스크립트 작성



- 서버는 사용자 패턴에는 관심 없음
 - 먼저 오는 순서대로 서비스 수행
- Peak구간 부하패턴을 재현하는 것이 중요함
 - 업무 별 부하량 파악이 중요

스크립트 작성



스크립트 작성 시 반복구간의 트랜잭션은

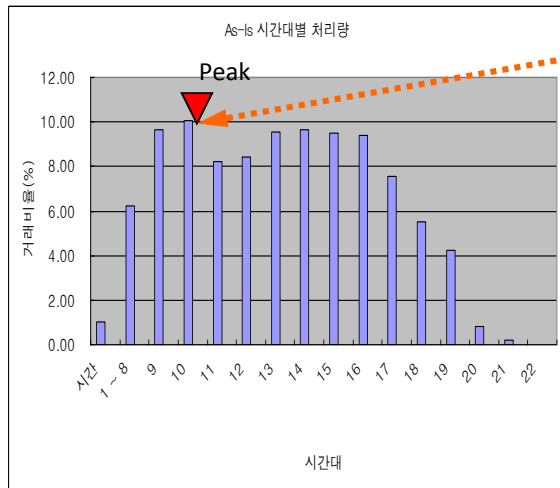
하나의 업무만 적용하여

스크립트를 분리하는 것이 합리적임

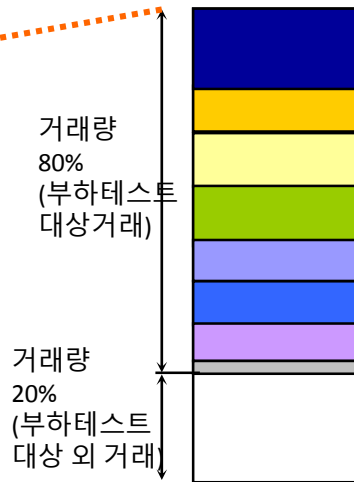
이렇게 해야 **Peak 구간의 업무 패턴을 재현**할 수 있음

스크립트 작성

단계1 As-Is Peak 시간대 분석



단계2 Url 정렬 (By 요청수)

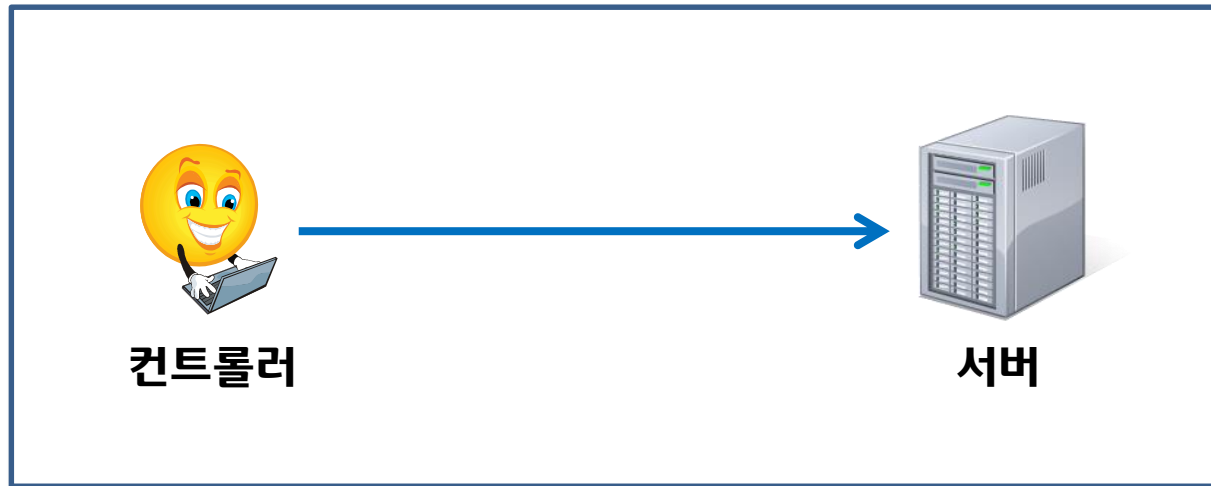


단계3 업무 별 TPS 정의

URL	TPS	업무명
a.Jsp	10	날짜_Bean
b.Jsp	9	고객검색
c.Jsp	8	개인고객조회
d.Jsp	7	Top 메뉴
e.Jsp	6	고객종합조회
f.Jsp	5	공통코드조회
g.Jsp	4	계약조회
h.jsp	3	메인화면

성능테스트 대상 업무만큼 스크립트를 작성해야 함.

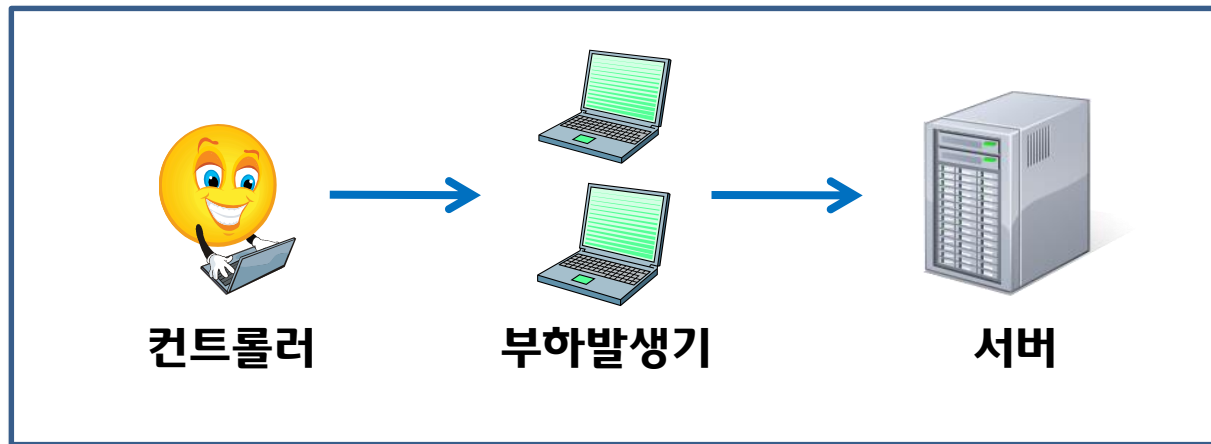
생각해보기 - 부하발생환경



위와 같이 컨트롤러 1대에서 성능테스트
한다면 어떤 상황이 발생할까요?

일반적인 부하 발생 환경

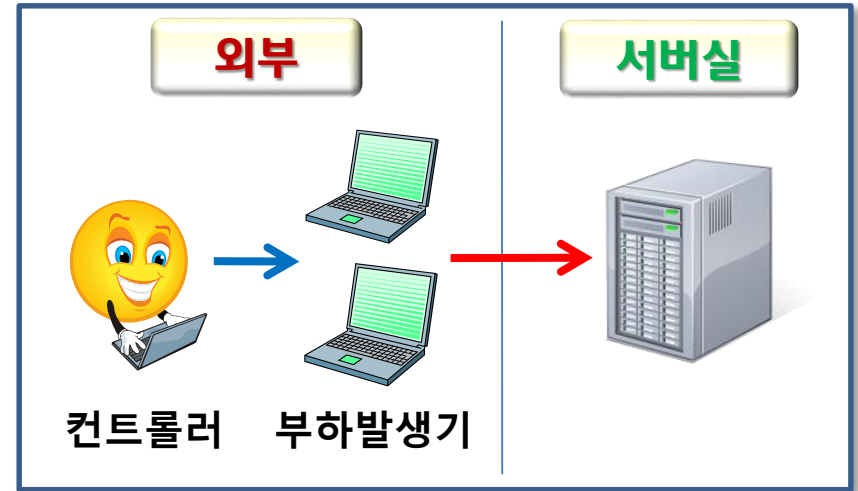
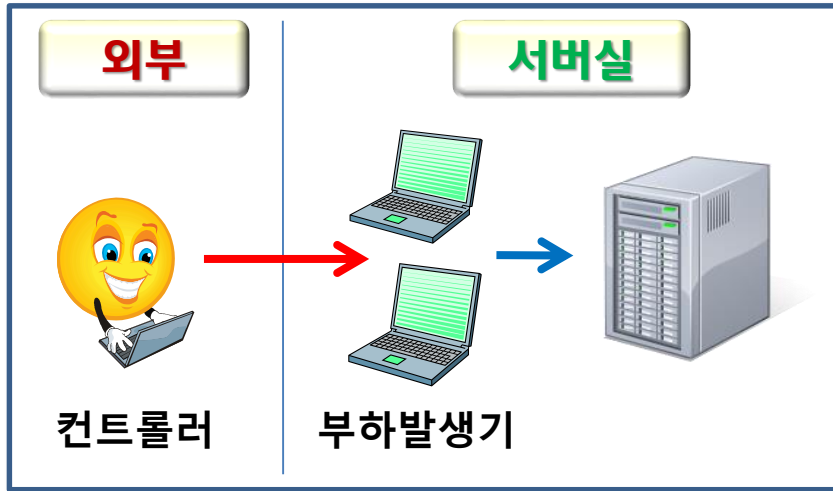
- 컨트롤러와 부하발생기를 분리



부하량이 많은 경우
컨트롤러 1대에서 부하발생하면
CPU, 메모리, Disk I/O, Network I/O 병목

일반적인 부하 발생 환경

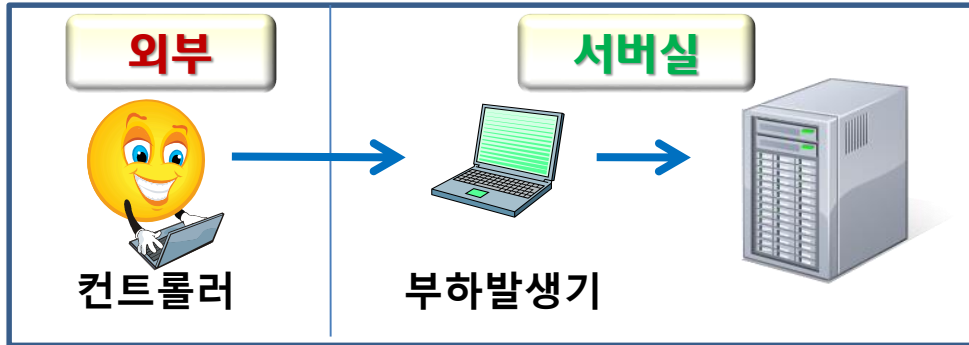
■ 부하발생기 위치



외부와 서버실 간의 Network 대역폭은 넓지 않음

그래서 서버실 외부에서 부하를 발생할 경우
Network 대역폭 부족으로 인한 **응답시간 지연 현상**이
발생할 수 있음

부하발생 환경 구성방안



컨트롤러와 부하발생기
사이에 **방화벽** 작업 필요



성능테스트 관련 인력이
전산센터로 이동해야 함



컨트롤러와 부하발생기
사이에 **방화벽** 작업 필요

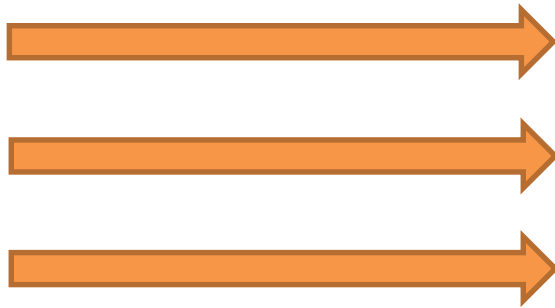
JMeter 부하발생 환경 구성

- Controller와 부하발생기의 설치파일은 동일
 - Controller : bin/jmeter.bat(sh) 실행
 - 부하발생기 : bin/jmeter-server.bat(sh) 실행
- Controller와 부하발생기 접속 형태
 - Server Port : 기본적인 정보 제공(def. 1099)
 - Server.rmi.localport : Sampler 수행(def. random)
 - Client.rmi.localport : 결과 데이터 전송(def. random)



생각해보기 – 서버 Resource 모니터링

서비스 요청



서버



서버 Resource 모니터링은
왜 해야 할까요?

서버 Resource 모니터링 이유

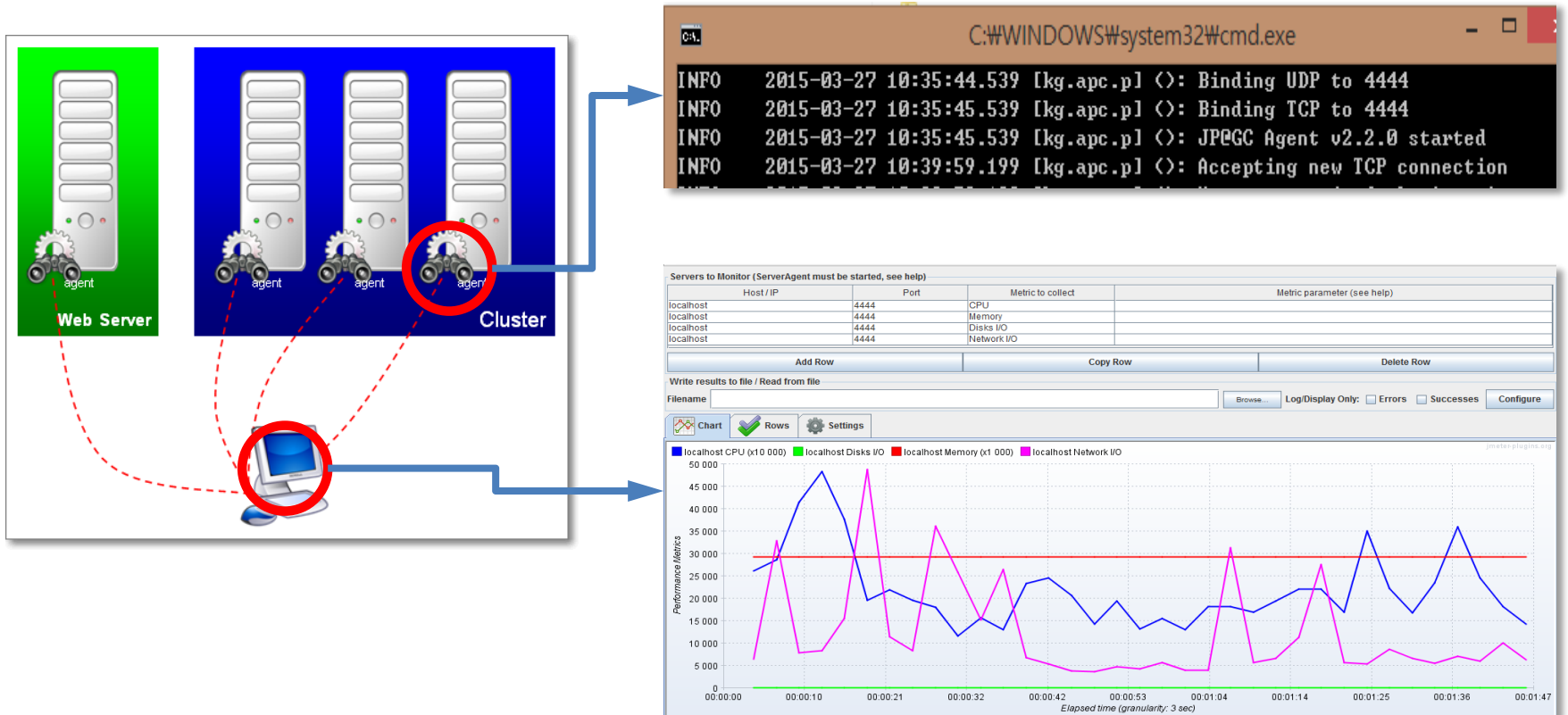
- 성능목표 검증
 - CPU 사용률은 성능목표에 반드시 포함됨
 - Single : 80%이하
 - Dual : 40%이하
- 튜닝 항목 도출
 - CPU 병목
 - Memory 병목
 - Disk 병목
 - Network 병목

서버 Resource 모니터링 개요

- 아래와 같은 시스템 서버 성능 모니터링 가능
 - Linux(x86,amd64,ppc,ppc64,ia64,s390,s390x)
 - Windows(x86,x64)
 - Solaris(sparc-32,sparc-64,x86,x63)
 - AIX(ppc,ppc64)
 - HP-UX(PA-RISC,ia64)
 - Mac OS(PowerPC,x86,x64)
- 아래와 같은 성능 지표 수집 가능
 - 총 75개의 성능 Metrics 지원
 - CPU, Memory, Disk I/O, Network I/O
 - JMX(gc-time, memory-usage 등)
 - TCP(established connections, time_wait, close_wait)
 - Swap(Used, Pagine, Pagout, free, total)
 - EXEC(Shell 파일을 직접 수행)

서버 Resource 모니터링 방법

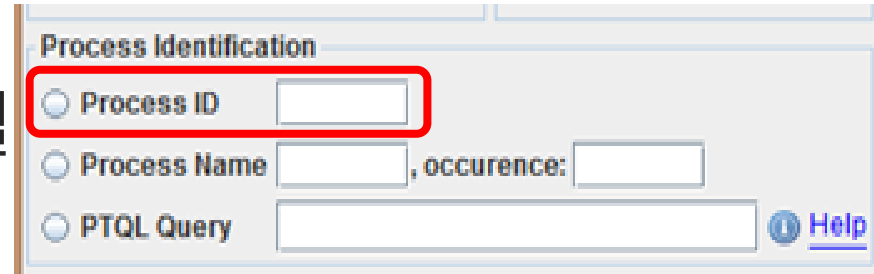
- Agent를 통한 서버모니터링 방식
 - 서버에 Agent를 설치



서버 모니터링 특징

■ 강력한 Filtering 기능

- CPU, 메모리
... 전체 or Process ID 별
사용량 측정가능



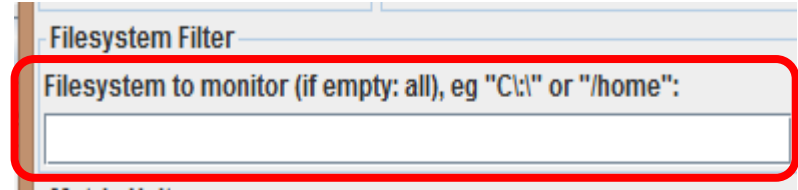
Process Identification

Process ID

Process Name , occurrence:

PTQL Query [Help](#)

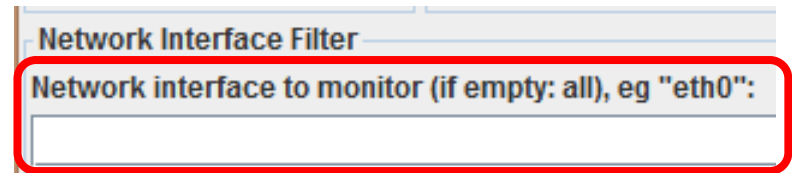
- Disk I/O
... FileSystem 별
Disk I/O 측정가능



Filesystem Filter

Filesystem to monitor (if empty: all), eg "C:\\" or "/home":

- NW I/O
... Network Interface 별
NW I/O 측정가능



Network Interface Filter

Network interface to monitor (if empty: all), eg "eth0":

서버 모니터링 특징

- EXEC를 이용하여 얼마든지 **모니터링 지표 확장** 가능
 - Unix : `"/bin/sh:-c:"` 로 시작
 - `/bin/sh:-c:free | grep Mem | awk '{print $7}'`
 - Windows : `cmd:/c:"` 로 시작
 - `cmd:/c: netstat -na | find "EST" /c`

jp@gc - PerfMon Metrics Collector

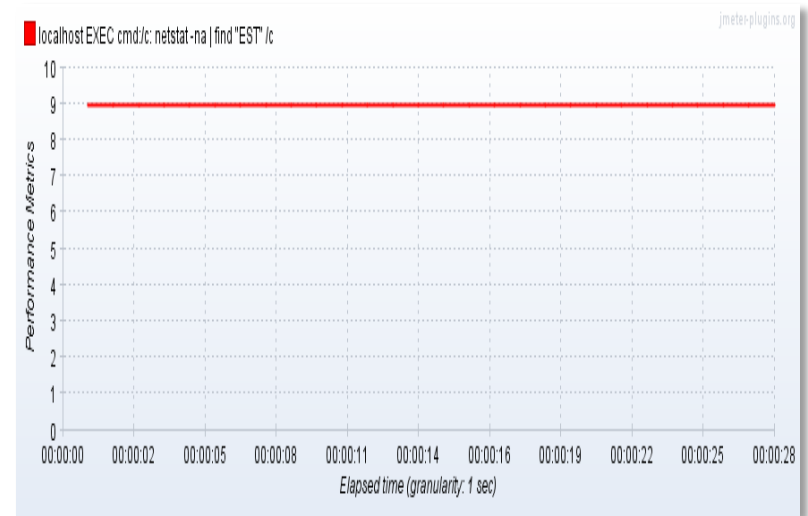
Name: jp@gc - PerfMon Metrics Collector

Comments:

[Help on this plugin](#)

Servers to Monitor (ServerAgent must be started, see help)

Host/IP	Port	Metric to collect	Metric name
localhost	4444	EXEC	cmd:/c: netstat -na find "EST" /c



생각해보기 – Workload 설계

- Think Time 없이 테스트 하는 경우

업무 명	호출 비율	업무 명	사용자수	업무 명	사용자수	응답시간	TPS	비율
A	50%	A	5	A	5	1	5	16%
B	30%	B	3	B	3	0.5	6	19%
C	20%	C	2	C	2	0.1	20	65%

Think Time 없이 테스트하면
응답시간에 영향을 받기 때문에
업무 비율을 맞출 수 없다.

생각해보기 – Workload 설계

$$\text{TPS} = (\text{사용자 수}) / (\text{호출간격})$$

$$\text{호출간격} = \text{응답시간} + \text{Think Time}$$

- Think Time 있는 상태에서 테스트

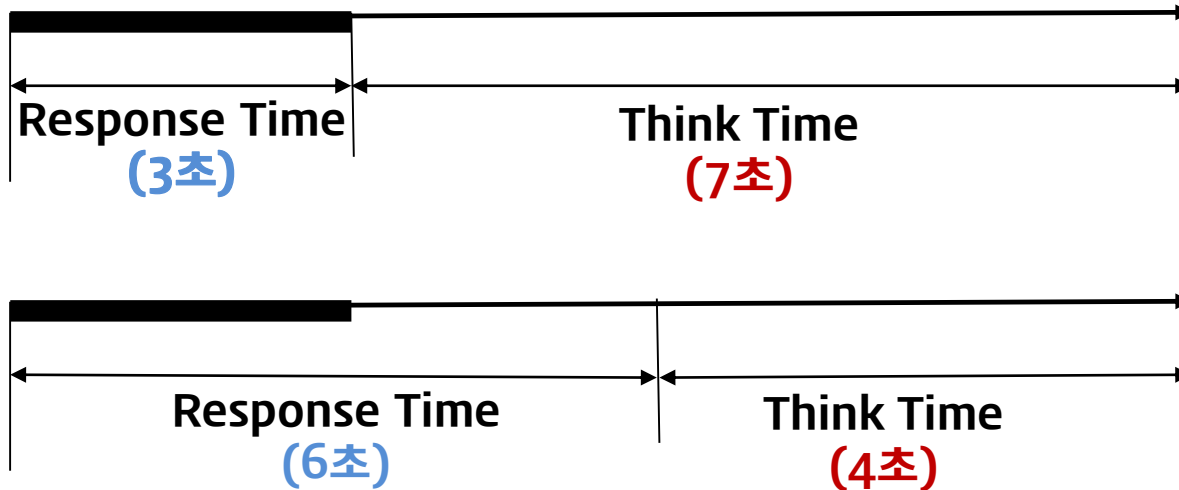
업무 명	목표 TPS	업무 명	1명 응답시간	업무 명	사용자수	Think Time	예상 TPS	실제 TPS
A	10	A	1	A	20	1	10	응답 시간에 따라 달라짐
B	20	B	1	B	40	1	20	
C	30	C	1	C	60	1	30	

부하량이 증가하면서 응답시간이 증가
정확한 호출간격을 제어할 수 없기 때문에
TPS를 제어할 수 없음

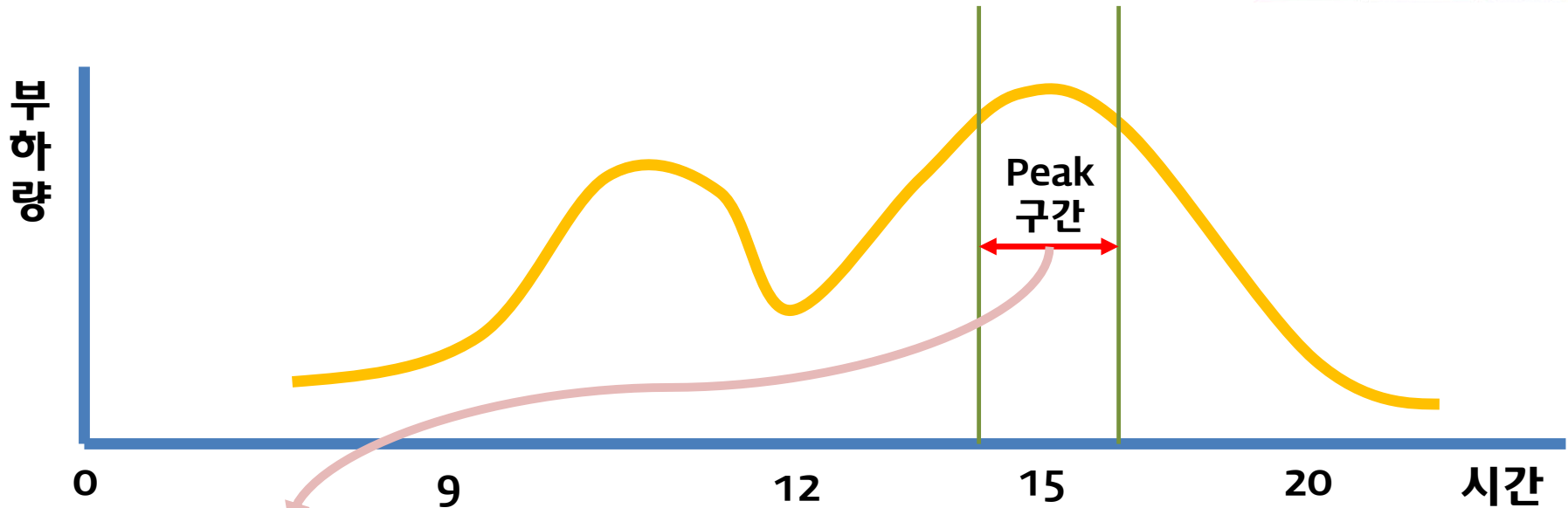
생각해보기 – Workload 설계

- 호출 간격을 제어
 - 응답시간에 따라 Think Time을 가변적으로 적용

호출 간격 = 10 초



WorkLoad 설계



업무 명	호출 건수 /1h	TPS
A	36,000	10
B	18,000	5
C	7,200	2
D	5,400	1.5
E	3,600	1
....

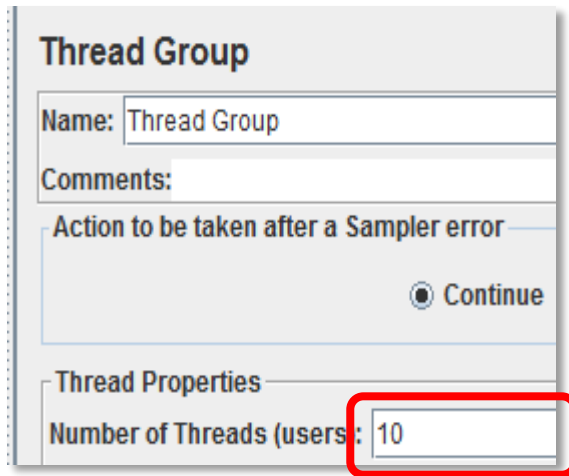


업무 명	사용자 수	호출 간격
A	50	5
B	10	2
C	20	10
D	15	10
E	5	5
....

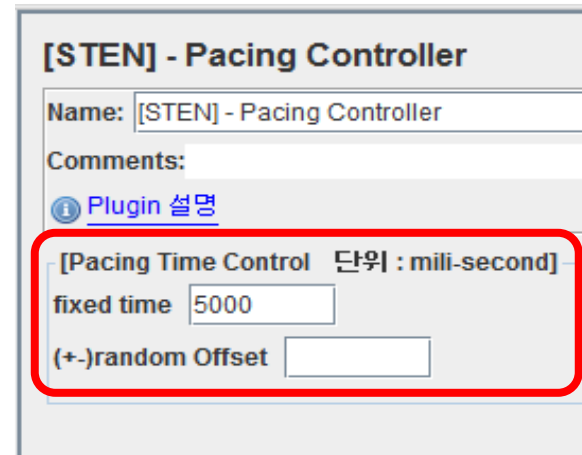
업무 명	사용자 수	호출 간격
A	100	10
B	50	10
C	20	10
D	15	10
E	10	10
....

JMeter WorkLoad 설계

- 업무 별 TPS 할당
 - 사용자 수 및 요청 간격 할당
 - 사용자수 : Thread Group
 - 요청간격 : STEN-Pacing Controller



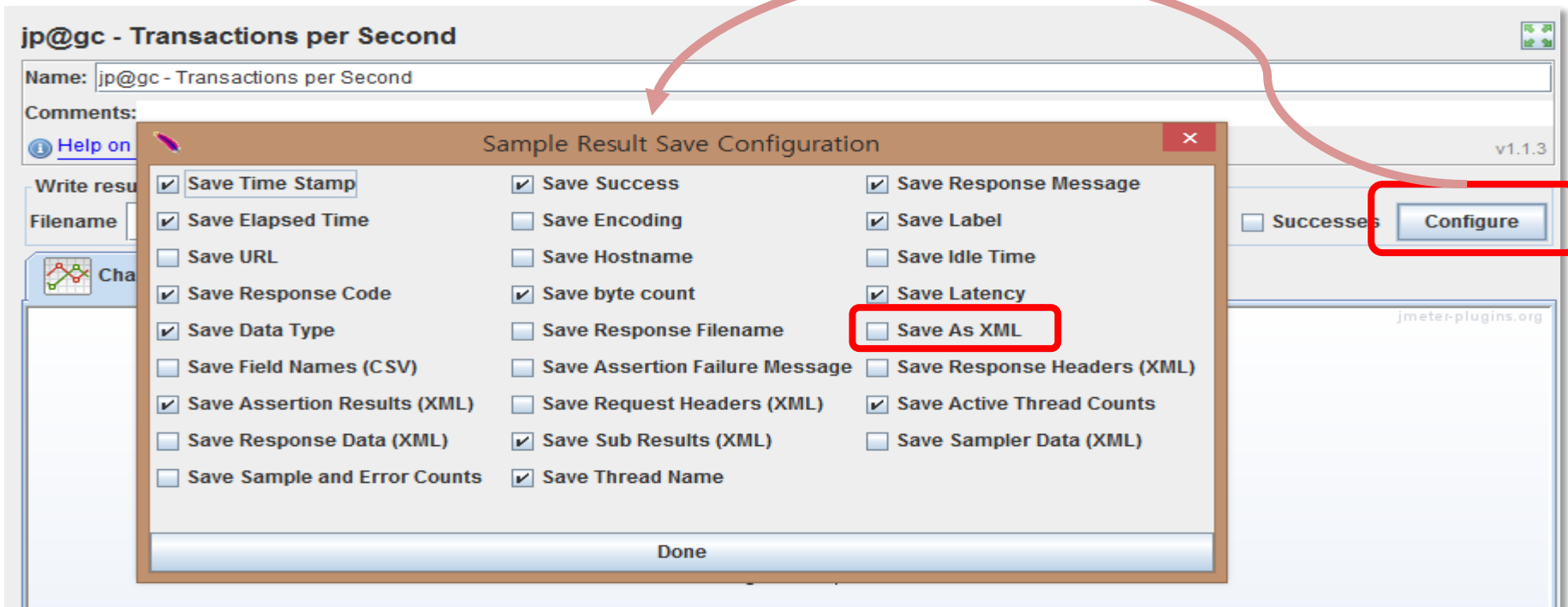
The screenshot shows the configuration for a Thread Group. The 'Name' field is set to 'Thread Group'. Under 'Action to be taken after a Sampler error', the 'Continue' radio button is selected. In the 'Thread Properties' section, the 'Number of Threads (users)' field is set to '10' and is highlighted with a red box.



The screenshot shows the configuration for a [STEN] - Pacing Controller. The 'Name' field is set to '[STEN] - Pacing Controller'. A link for 'Plugin 설명' is visible. The 'fixed time' field is set to '5000' and is highlighted with a red box. The '(+)-random Offset' field is empty.

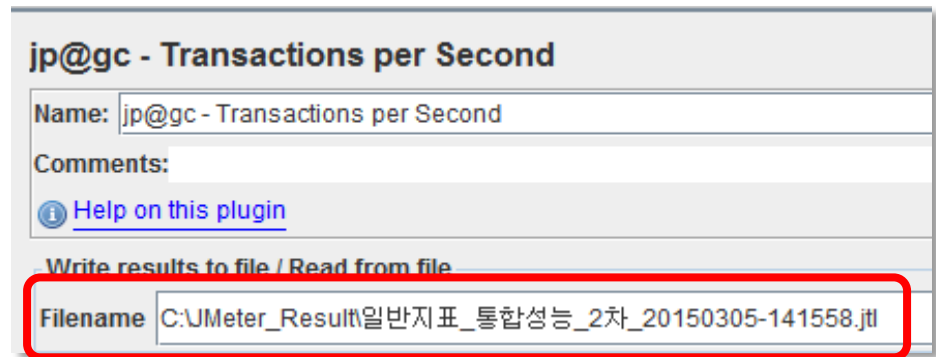
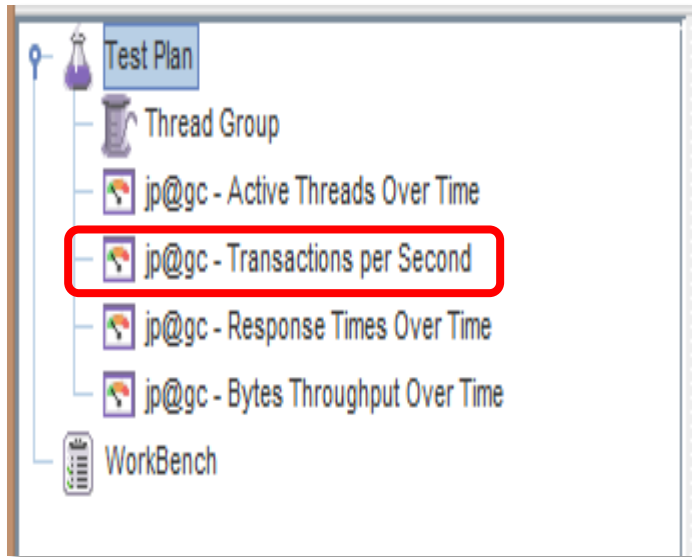
결과 데이터 저장 방법

- 테스트 결과 분석을 위해 데이터를 저장
 - CSV 와 XML 방식



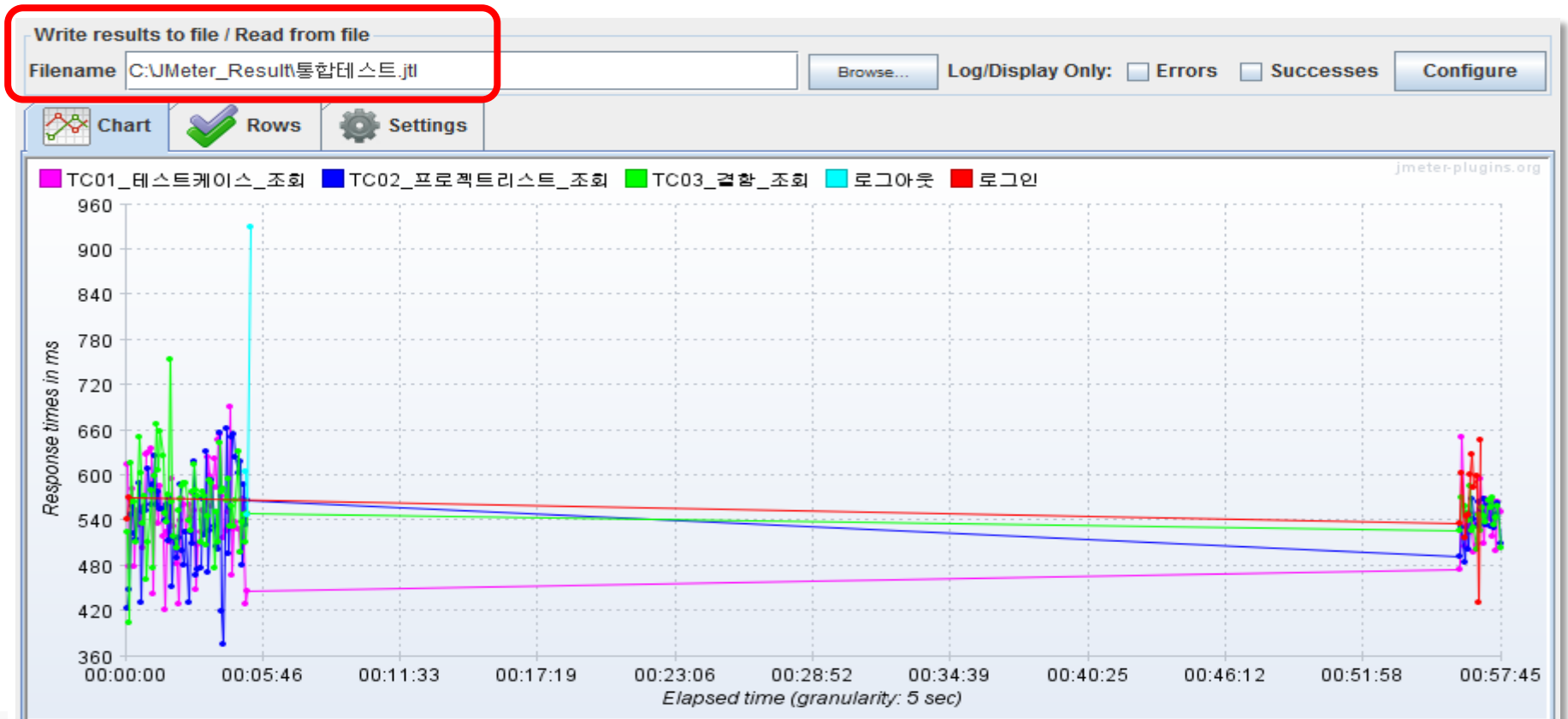
결과 데이터 저장 방법

- 하나의 Listener에 대표로 파일을 저장
 - 공통 Format이므로 Listener 별로 저장할 필요 없음



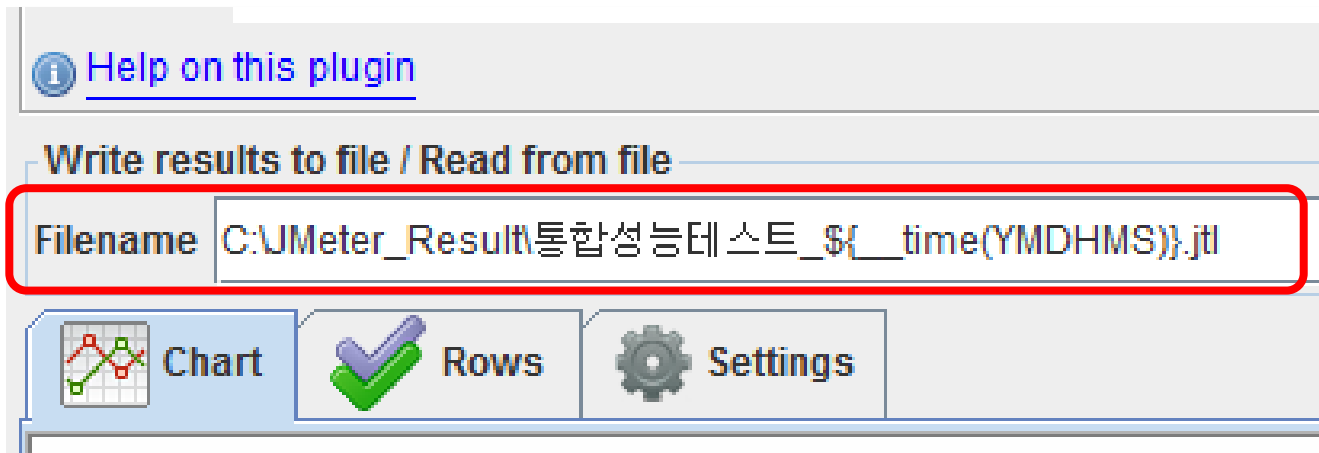
결과 데이터 저장 방법

- 같은 파일로 결과 데이터를 남기면 기본적으로 Merge 됨



결과 데이터 저장 방법

- Merge 피하는 방법
 - 테스트 할 때마다 파일 이름 변경
 - 파일이름에 내장함수 사용
 - `${_time(YMDHMS)}` 사용
 - 수행할 때마다 파일이름이 다름



내장함수 생성 방법

- Package 이름이 functions를 포함해야 함
- AbstractFunction 클래스 상속 및 아래 함수 구현
 - Execute
 - setParameters
 - getReferenceKey
 - getArgumentDecs
- Jar 파일을 생성 <JMETER_HOME>/lib/ext에 위치시킴

내장함수 생성 방법

```
package my.jmeter.functions;
```

```
import java.util.Collection;
```

```
public class FunctionTemplate extends AbstractFunction
```

```
@Override
```

```
public List<String> getArgumentDesc() {  
    // TODO Auto-generated method stub  
    return null;  
}
```

```
@Override
```

```
public String execute(SampleResult arg0, Sampler arg1)  
    throws InvalidVariableException {  
    // TODO Auto-generated method stub  
    return null;  
}
```

```
@Override
```

```
public String getReferenceKey() {  
    // TODO Auto-generated method stub  
    return null;  
}
```

```
@Override
```

```
public void setParameters(Collection<CompoundVariable> arg0)  
    throws InvalidVariableException {  
    // TODO Auto-generated method stub  
}
```

```
}
```

내장함수 확장성

- 인터넷 뱅킹과 같은 암호호화 업무 스크립트 작성 가능
 - Java로 된 API 함수 받아서 구현
 - dll로 제공될 경우 JNI 나 JNA 기술 이용하여 구현

- 전문가 의견 소개

- <http://blazemeter.com/blog/load-testing-challenges-present-future-interview-alex-podelko>

Alex Podelko is a prominent thought leader in the world of load and performance testing. For the **past sixteen years**, Alex has supported numerous performance initiatives for several large companies. He is currently a **consulting member of technical staff at Oracle**

Q: How do you see the future of load testing?

In my opinion, the load test process **needs to undergo a change in culture**. Traditionally **the load testing team joined** the development process only **at the very end**. A better solution, however, might be to introduce load testing into the process **from the beginning**, making sure that every part of the application can withstand the anticipated load.

One solution is to make the testing **process as automated as possible**. This way the performance engineers only **need to be involved with complex cases**. Another important change may be to give more of the **load testing responsibilities to the developers and functional testers**