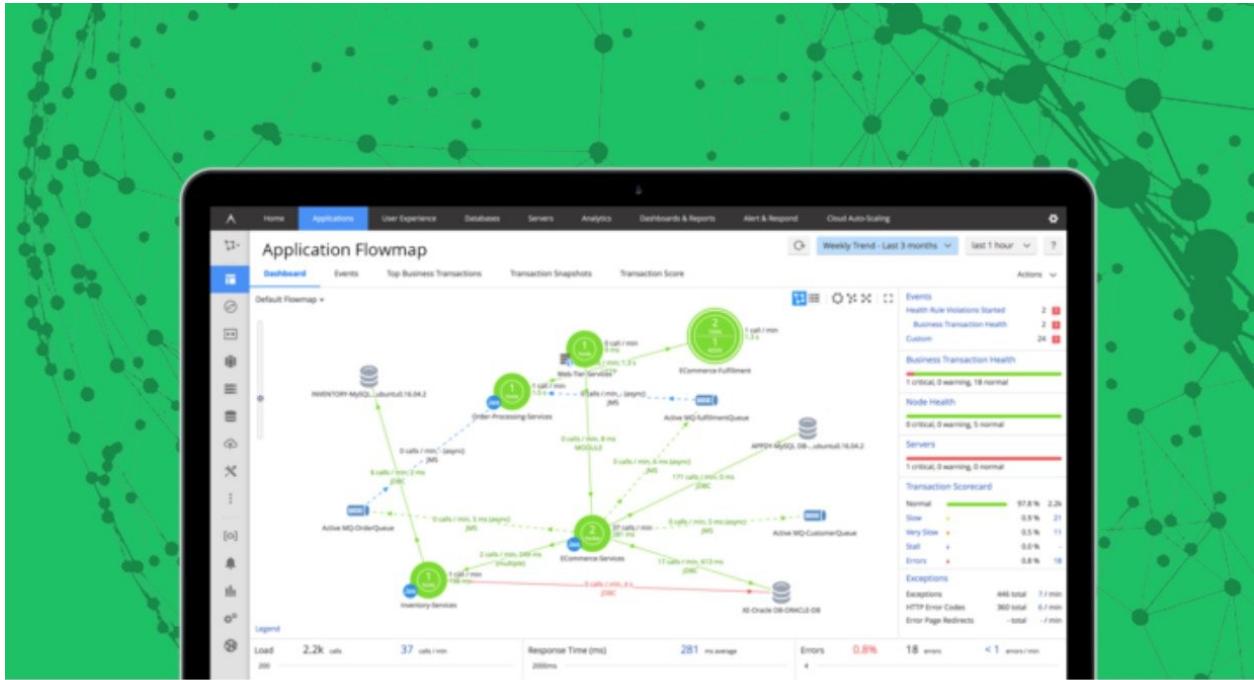


AppDynamics Lab exercises

Java Lab



Notes

Introduction

Login Ravello, <https://login.ravellosystems.com>



jules.weijters@gmail.com

Password

+ Identity Domain

[Forgot your password?](#)

LOGIN

Start VM, when started.

Java 101 Template

Standard CentOS 6.3. Log-in with the user ravello (using...
[more...](#)

2 CPUs | 8 GB Memory | 32 GB Total Storage

1 Network Connection

- Nic #0: Inbound & Outbound (1 External IP)

VM is Started

External access for: Nic NIC1 10.0.0.3

IP: 129.213.56.249

Ports: 8090 (AppD) 80 (http) 22 (ssh) [Open](#)

SSH Shell access

iTerm profile !_AppDynamics Lab App

```
ssh -p 22 ravello@129.213.27.78
```

Start Tomcat application:

```
sudo su - tomcat
export TOMCAT_HOME=/opt/tomcat/apache-tomcat-8.0.30/
$TOMCAT_HOME/bin/catalina.sh start
```

Application Access

Tomcat:

<http://java101template-candidatejwjava101-lgflyzv1.srv.ravcloud.com>

Sample App:

http://java101template-candidatejwjava101-lgflyzv1.srv.ravcloud.com/Cars_Sample_App/home.do

AppDynamics Access

<https://login.appdynamics.com/sso/authenticate/?site= selfservice&target=https://portal.appdynamics.com/account/subscriptions/>

Subscriptions

Select a subscription (<https://kabuki201709280411126.saas.appdynamics.com>)

SAAS - kabuki201709...
EXP 10/18/2017

[Start On-Premises Trial >](#)

Subscription Details

Expires in **14 Days**

on Oct. 18, 2017, midnight

[Show Controller Login Info](#)

[Launch AppDynamics](#)

Excercise 1: Monitoring Installation

What do you want to monitor?

Applications



User Experience



2 JVM

Sun/JRockit

3 Configure

We've populated the connection information for this Controller for you. If you want to use another Controller, check your welcome email from AppDynamics for the host and port if you signed up for a SaaS account.

`https:// kabuki201709280411126.saa : 443` Use SSL

5 Install the App Agent

1. Unzip the Java Agent. For example:

```
unzip AppServerAgent-4.3.3.zip -d <java_agent_unzip_path>/appagent
```

2. Edit the startup script for your application server. Add the -javaagent parameter to the startup command.

Set the value of -javaagent to the path to the Java Agent JAR file. For example:

```
-javaagent:<java_agent_unzip_path>/appagent/javaagent.jar
```

3. Restart your application

For exact instructions, please select your Application Server:

Apache Tomcat

[Apache Tomcat - Install Instructions](#)

<https://docs.appdynamics.com/display/PRO43/Apache+Tomcat+Startup+Settings>

Unzipped agent to /home/tomcat/appagent owned by tomcat:tomcat

Created /opt/tomcat/apache-tomcat-8.0.30/bin/setenv.sh with
export CATALINA_OPTS="\$CATALINA_OPTS -javaagent:/home/tomcat/appagent/javaagent.jar"

chmod +x /opt/tomcat/apache-tomcat-8.0.30/bin/setenv.sh

Update agent config XML

vi /home/tomcat/appagent/conf/controller-info.xml

Updated nodes:

```
<application-name>Tomcat Sample App</application-name>
<tier-name>TomcatAppServer</tier-name>
<node-name>JRocket</node-name>
```

Restarted Tomcat:

```
[tomcat@Java101Template bin]$ ./shutdown.sh
Using CATALINA_BASE: /opt/tomcat/apache-tomcat-8.0.30
Using CATALINA_HOME: /opt/tomcat/apache-tomcat-8.0.30
Using CATALINA_TMPDIR: /opt/tomcat/apache-tomcat-8.0.30/temp
Using JRE_HOME: /usr
Using CLASSPATH: /opt/tomcat/apache-tomcat-8.0.30/bin/bootstrap.jar:/opt/tomcat/
apache-tomcat-8.0.30/bin/tomcat-juli.jar

[tomcat@Java101Template bin]$ ./startup.sh
Using CATALINA_BASE: /opt/tomcat/apache-tomcat-8.0.30
Using CATALINA_HOME: /opt/tomcat/apache-tomcat-8.0.30
Using CATALINA_TMPDIR: /opt/tomcat/apache-tomcat-8.0.30/temp
Using JRE_HOME: /usr
Using CLASSPATH: /opt/tomcat/apache-tomcat-8.0.30/bin/bootstrap.jar:/opt/tomcat/
apache-tomcat-8.0.30/bin/tomcat-juli.jar

Tomcat started.
```

Make sure agent is running in /opt/tomcat/apache-tomcat-8.0.30/logs/catalina.out

Generate load and check if logfile created in /home/tomcat/appagent/v

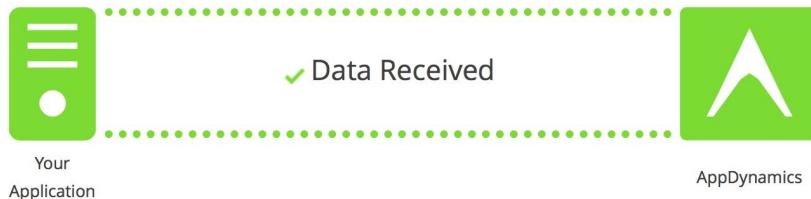
As user Ravello run load:

```
[ravello@Java101Template ~]$ ./generate-load.sh
```

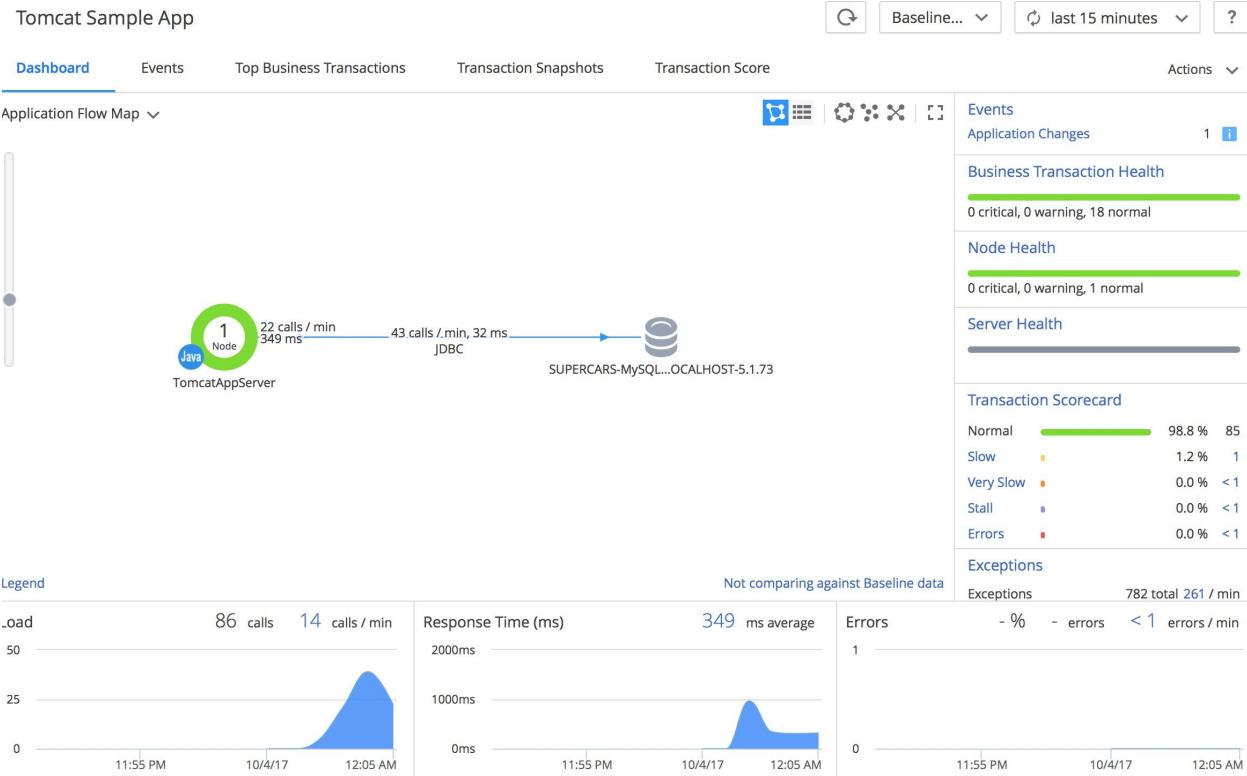
6 Connect the Agent to the AppDynamics Controller

Generate load on your instrumented application.

✓ Agent Connected



Continue



Exercise 2: Business Transaction configuration

Open Configuration/Transaction Detection/Rules

The screenshot shows the AppDynamics Applications interface. The left sidebar has a 'Configuration' section with 'Rules' selected. The main area is titled 'Instrumentation' under 'Transaction Detection'. It lists several auto-discovery rules:

- Java Auto Discovery Rule (Default Scope)
- .NET Auto Discovery Rule (Default Scope)
- Web Server Auto Discovery Rule (Default Scope)
- Node.js Auto Discovery Rule (Default Scope)
- PHP Auto Discovery Rule (Default Scope)
- Python Auto Discovery Rule (Default Scope)
- Quartz (Default Scope)
- Spring WS - dispatching ... (Default Scope)
- Struts Action Servlet (Default Scope)
- JBoss web-services servlet (Default Scope)
- Spring WS - Base servlet ... (Default Scope)

Double-click Java - Java Auto Discovery rule and clear Struts Action. Save (works however without excluding Struts from default).

The screenshot shows the 'Rule Editor' for the 'Java - Java' rule under 'Instrumentation' in the 'Rule Configuration' tab. The 'Binary Remoting' section has 'Transaction Monitoring Enabled' checked and 'Discover Transactions automatically for all Binary Remoting requests (Thrift)' checked. The 'Servlet' section has 'Transaction Monitoring Enabled' checked and 'Discover Transactions automatically for all Servlet requests' checked. The 'Struts Action' section has both 'Transaction Monitoring Enabled' and 'Discover Transactions automatically for all Struts Action invocations' unchecked.

Double-click Java - Struts Action Servlet exclude rule and uncheck Enabled. Save.

Rule Editor

Summary Rule Configuration

Rule Type	Custom Match Rule
Include/Exclude	<input type="radio"/> <input checked="" type="radio"/> Include Transactions discovered by this rule <input checked="" type="radio"/> <input type="radio"/> Exclude Transactions discovered by this rule
Agent Type	<input checked="" type="radio"/> Java
Entry Point Type	<input checked="" type="radio"/> Servlet
Name	Struts Action Servlet
Enabled	<input type="checkbox"/>
Priority	0
Description	Struts Action Servlet
Scope	Default Scope
Created By system at 10/04/17 12:00:55 AM	
Last Modified By	jules.weijters at 10/04/17 9:42:22 AM

Generate load and check Business Transactions. Only .do Transactions are shown.

Tomcat Sample App

- Application Dashboard
- Business Transactions**
- Service Endpoints
- Tiers & Nodes
- Servers
- Database Calls
- Remote Services
- Troubleshoot
- More

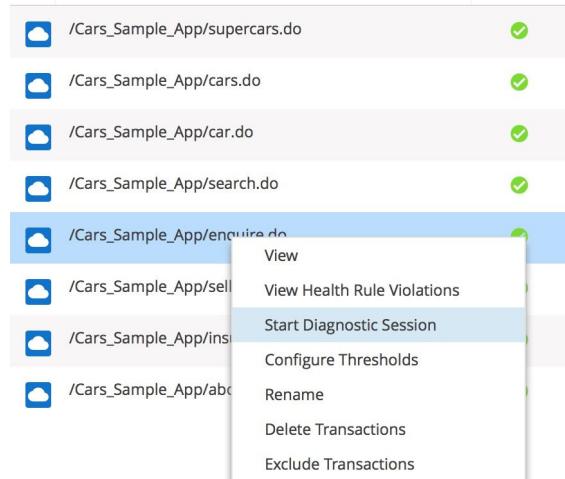
Information Pages

Business Transactions

Name	Health
/Cars_Sample_App/supercars.do	✓
/Cars_Sample_App/cars.do	✓
/Cars_Sample_App/car.do	✓
/Cars_Sample_App/search.do	✓
/Cars_Sample_App/enquire.do	✓
/Cars_Sample_App/sell.do	✓
/Cars_Sample_App/insurance.do	✓
/Cars_Sample_App/about.do	✓

Exercise 3: Diagnostic Sessions

Right-click `enquire.do` to start a Diagnostics Session manually.



Leave defaults, click start and generate load.

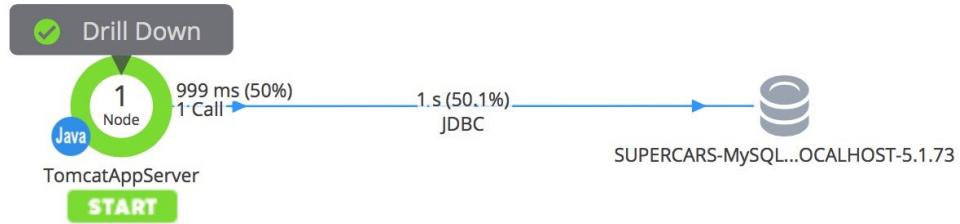
Go back to Application Dashboard and find Transaction Snapshots/Diagnostics Sessions.

Type	Time	Summary	Exe Time ...	Business Transaction
▶ U Manual	10/04/17 10:20:28 AM	Starting Diagnostic Session for b...	-	/Cars_Sample_App/enquire.do

Check if there's multiple sessions for enquiry.do. In this case there's the manual session and an triggered session because of Slow rate.

Type	Time	Summary	Exe Time ...	Business Transaction
▶ U Manual	10/04/17 10:20:28 AM	Starting Diagnostic Session for b...	-	/Cars_Sample_App/enquire.do
▶ U Abnormal Slow Rate	10/04/17 10:01:48 AM	Starting Diagnostic Session after...	-	/Cars_Sample_App/sell.do
▶ U Abnormal Slow Rate	10/04/17 9:53:48 AM	Starting Diagnostic Session after...	-	/Cars_Sample_App/search.do
▶ U Abnormal Slow Rate	10/04/17 9:53:48 AM	Starting Diagnostic Session after...	-	/Cars_Sample_App/enquire.do

Compare the calls, take a well performing and the worst performing from the sessions. Use View Snapshot to drill into the snapshot.



The slower of the two is actually doing a SQL query saving enquiry data.

Comparing URLs seeing differentiation in the query string, /Cars_Sample_App/sell.do?query=save.

For enquiry, where the save action is doing the query:

/Cars_Sample_App/enquire.do?car=608&carName=s%A0DB9

/Cars_Sample_App/enquire.do?query=save

Changed the servlet naming ruling to use the value from parameter ?query. Works but not great.

A screenshot of the 'Rule Editor' in an APM tool. The left sidebar shows 'Instrumentation' and 'Transaction Detection' sections with various agent types like Java, .NET, Web Server, etc. The main panel is titled 'Rule Configuration' under the 'Summary' tab. It contains two sections: 'Binary Remoting' and 'Servlet'. In the 'Binary Remoting' section, 'Transaction Monitoring Enabled' and 'Discover Transactions automatically for all Binary Remoting requests (Thrift)' are checked. In the 'Servlet' section, 'Transaction Monitoring Enabled' and 'Discover Transactions automatically for all Servlet requests' are also checked. Under 'Configure Naming' for Servlet, there is a dropdown 'Use the first' set to '2 segments of the URL' with a note '(1)', and a checked checkbox 'Name Transactions dynamically using part of the request'. Below this, a dropdown 'Use' is set to 'a parameter value' with a note '(1)' and a text input 'Parameter Name' containing 'query'.

Save, run load. Result should show when call includes query resulting in distinction between transactions.

	/Cars_Sample_App/sell.do.save	✓
	/Cars_Sample_App/sell.do	✓
	/Cars_Sample_App/search.do.search	✓
	/Cars_Sample_App/search.do	✓
	/Cars_Sample_App/insurance.do	✓
	/Cars_Sample_App/enquire.do.save	✓
	/Cars_Sample_App/enquire.do	✓

Exercise 4: SQL Parameter Capture

Currently no bind variables in SQL capture.

[SQL Calls](#) [Remote Service Calls](#)

SQL Calls Summary for TomcatAppServer

Query Type	Query
Insert	INSERT INTO ENQUIRIES (NAME, EMAIL, COMMENT, CAR_ID, DUMMY) SELECT ?,?,?,?, SLEEP(1)

Change SQL Capture Settings via Configuration/Instrumentation/Call Graph Settings.

NOTE: Issue with the scrollong on the page in multiple browsers!! Can't reach save at the bottom unless huge screensize.

Call Graph Settings

Minimum SQL query time: 10 milliseconds (minimum value: 10 ms). If a query takes longer than this, it will be captured along with the calling method.

Packages to Exclude from Call Graphs

Since a call graph can potentially have hundreds of methods, to make it more readable and to ensure low overhead, certain packages can be excluded. These are typically Java libraries, and App Server container classes. The packages marked as "Required" below cannot be removed from the Excluded Packages list, however you can add and remove other packages.

Excluded Packages	Description
java.*	Core Java Libraries
javax.*	Java Extension/J2EE Libraries
com.sun.*	Sun Reference Implementations/JDK Core Classes
sun.*	Sun Reference Implementations/JDK Core Classes
org.*	Common Open Source Libraries like Apache's
com.bea.*	BEA Core Classes

Add Custom Package Exclude | Remove Selected

Specific sub-packages / classes from the Excluded Packages to be included in call graphs.

To force inclusion of specific classes or sub-packages from the Excluded Packages list above, specify them below. For example, java.* is in the Excluded Packages list, but the list below.

Packages/Classes Always Shown	Description
org.apache.kafka.*	Apache Kafka Classes
org.apache.camel.example.*	Apache Camel example classes
org.apache.camel.component.seda.*	Apache Camel Seda internals
org.mule.processor.*	Mule ESB classes
org.jboss.soa.esb.client.*	JBoss ESB classes
org.springframework.integration.*	Spring Integration classes

Add Always Shown Package/Class | Remove Selected

SQL Capture Settings

Capture Raw SQL - This will capture raw SQL statements with parameter values included. For example "select * from user where user_SSN = '111-11-1111';"

Save and generate load. Now check to see bind variables.

Exercise 5: Custom Error Detection

Diagnostics session enabled for /Cars_Sample_App/sell.do.save being the source of the XMLError shown in the catalina.out log. Not all transactions have an exception find one and right-click to create an Information Point on the Logging.Logger.Log call. This shows the input for the custom error detection.

Execution Time 794 ms Wait Time 773 ms Block Time

Details Filters Set Root Reset Root Export

Name

- Servlet - action:doPost
- Servlet - action:process:1913
 - Struts Action - ActionSell:execute:45
 - Struts Action - ActionSell:saveCarForm:56
 - supercars.dataloader.CarDataLoader:saveCar:5
 - supercars.logging.Logger:log:23
 - supercars.XMLException:printStackTrace:1
- HTTPServlet:service:729

Add Information Point

Specify Name and Agent Type for this Info Point

Name:

Agent Type: Java .NET PHP

Define the Method Signature

Specify the **Java method signature** with the fully qualified class/interface/super-class/annotation name and the method name.

Class: with a Class Name that: equals: supercars.logging.Logger

Method Name: log

Method Parameters (optional): Is this Method Overloaded?

Add Parameter

Next add error logging to Configuration/Instrumentation/Error Detection.

Instrumentation

Transaction Detection Backend Detection **Error Detection** Service Endpoints Data Collectors Call Graph Settings JMX

Error Detection [?](#)

[Save Error Configuration](#)

Error Detection Using Logged Exceptions or Messages

▼ Define where AppDynamics will look for log messages or exceptions to detect errors

Detect errors logged using java.util.logging (Java 1.6 is required)

Detect errors logged using Log4j

Detect errors logged using SLF4j/Logback

Detect errors by looking at messages logged with ERROR or higher severity

Mark Business Transaction as error

Configure AppDynamics to look for logged errors or exceptions in any use a custom logger other than log4j or java.util.logging.

Name

Add Custom Logger Definition [Edit](#) [Delete](#)

Click Add Custom Logger Definition and add the found Class and Method.

Custom Logger Definition

A custom logger definition needs the method signature with the fully qualified class/interface/super-class/ann method name. If the method is overloaded, it is recommended that you also specify the fully qualified paramet

Name Enabled

Define the Class and Method Signature

Class equals

Method Name

Method Parameters

[Add Parameter](#)

Specify the parameter index for the exception object

Exception Parameter

Add a parameter as `java.lang.Exception`, click Save.

Now generate load and observe errors coming in. Drill-down into the Business Transaction with errors to find the full Exception.

Transaction: b79a27a4-253a-4634-a494-98e45653e18c | 603 ms - TomcatAppServer | X

Overview	Slow Calls & Errors	DB & Remote Service Calls	Server	Data Collectors	More
Summary	Slowest Calls and Errors				
User Experience Error Execution Time 603 ms CPU Time 2.5 ms Timestamp 10/05/17 10:34:18 PM (server) 10/05/17 10:34:18 PM (agent)	No Slow Calls Found Error supercars.XMLException XML Example Exception Thrown at supercars.dataloader.CarDataLoader.saveCar(CarDataLoader.java:51) at supercars.action.ActionSell.saveCarForm(ActionSell.java:56) at supercars.action.ActionSell.execute(ActionSell.java:45) at org.apache.struts.chain.commands.servlet.ExecuteAction.execute(ExecuteAction.java:58) at org.apache.struts.chain.commands.AbstractExecuteAction.execute(AbstractExecuteAction.java:67) at org.apache.struts.chain.commands.ActionCommandBase.execute(ActionCommandBase.java:51) at org.apache.commons.chain.impl.ChainBase.execute(ChainBase.java:191) at org.apache.commons.chain.generic.LookupCommand.execute(LookupCommand.java:305) at org.apache.commons.chain.impl.ChainBase.execute(ChainBase.java:191) at org.apache.struts.chain.ComposableRequestProcessor.process(ComposableRequestProcessor.java:283) at org.apache.struts.action.ActionServlet.process(ActionServlet.java:1913) at org.apache.struts.action.ActionServlet.doPost(ActionServlet.java:462) at javax.servlet.http.HttpServlet.service(HttpServlet.java:648) at javax.servlet.http.HttpServlet.service(HttpServlet.java:729) at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:291) at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:206) at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:52) at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:239) at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:206) at org.apache.catalina.core.StandardWrapperValve.invoke(StandardWrapperValve.java:212) at org.apache.catalina.core.StandardContextValve.invoke(StandardContextValve.java:106) at org.apache.catalina.core.StandardAuthenticator.authenticate(Base.java:502) at org.apache.catalina.core.StandardHostValve.invoke(StandardHostValve.java:141)				
Node JRocket Tier TomcatAppServer Business Transaction /Cars_Sample_App/sell.do.save URL /Cars_Sample_App/sell.do <input type="button" value="🔍"/>					
More Details >					

Exercise 6: Information Points

Car save found in CarDataLoader.java

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights ▾

Branch: 1.0-MAINTENANCE ▾ Cars_Sample_App / src / main / java / supercars / dataloader / CarDataLoader.java

Start Diagnostics Session for Business Transaction /Cars_Sample_App/sell.do.save. Enter a sell into the webapp and search the diagnostics session. Find the supercars.dataloader.CarDataLoader:saveCar call. Right-click and select Configure Instrumentation for this Class/Method.

Transaction: e743ac1d-e882-48cd-8331-57b58298f9d8 | 68 ms - TomcatAppServer X

Overview Call Graph Slow Calls & Errors DB & Remote Service Calls Server Data Collect

Execution Time 67 ms Wait Time 46 ms Block Time 0 ms Node JRocket Timestamp 10/06/17 12:50:00

Details Filters Set Root Reset Root Export

Name

- ▼ Servlet - action:doGet
 - ▼ Servlet - action:process:1913
 - ▼ Struts Action - ActionSell:execute:45
 - ▼ Struts Action - ActionSell:saveCarForm:56
 - ▼ supercars.dataloader.CarDataLoader:saveCar:53
 - ▼ supercars.logging.Logger:log:23
 - supercars.XMLEception:printStackTrace:28
 - ▼ HttpServlet:service:729
 - ▼ HttpServlet:service:729
 - ▼ JSPBaseServlet:service:70

- View Details
- Copy to Clipboard
- Collapse
- Expand All
- Set Root
- Reset Root
- Configure Instrumentation for this Class/Method

Select Information Point.

Configure Instrumentation

What type of Instrumentation Configuration would you like to create for CarDataLoader.saveCar?

Type

Set name and add Average and Sum.

Edit Information Point - Car Prices

Specify Name and Agent Type for this Info Point

Name

Agent Type Java .NET PHP

Define the Method Signature

Specify the **Java method signature** with the fully qualified class/interface/super-class/annotation name and the method name. If the method is **overloaded** you can specify the parameters.

Class equals

Method Name Is this Method Overloaded?

Method Parameters (optional...)

Match Conditions (optional)

Define Custom Metrics using the Method Parameters or Return Value

Name	Custom Metric
------	---------------

Price Average	AVERAGE (ParamIndex_0.getPrice)
---------------	---------------------------------

Price SUM	SUM (ParamIndex_0.getPrice)
-----------	-----------------------------

Use getPrice to get the price from the Carform class in the Use Getter Chain option.

Information Point Custom Metric Definition

X

Custom business metrics can be generated from the Java method invocation using the values of the parameters/return values.

To configure custom metrics, pick the parameter index or return value and an optional getter operation on it if it is a complex object.

Name

Data To Collect

Collect Data From Method Parameter @ Index:

Return Value

Invoked Object

Operation on Method Parameter Use `toString()`

Use Getter Chain

for example: `getAccount().getBalance()`

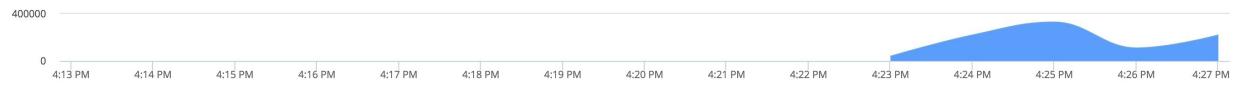
[more help](#)

Generate load and observer Applications/Information Points to see the metrics show up.

Custom Metric - Price Sum

920000 sum

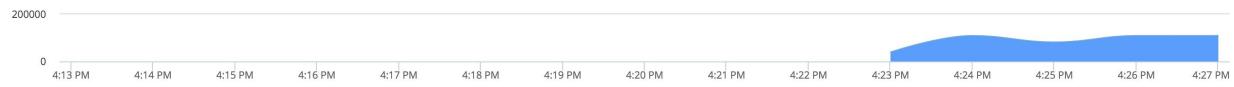
Custom Metric - Price Sum



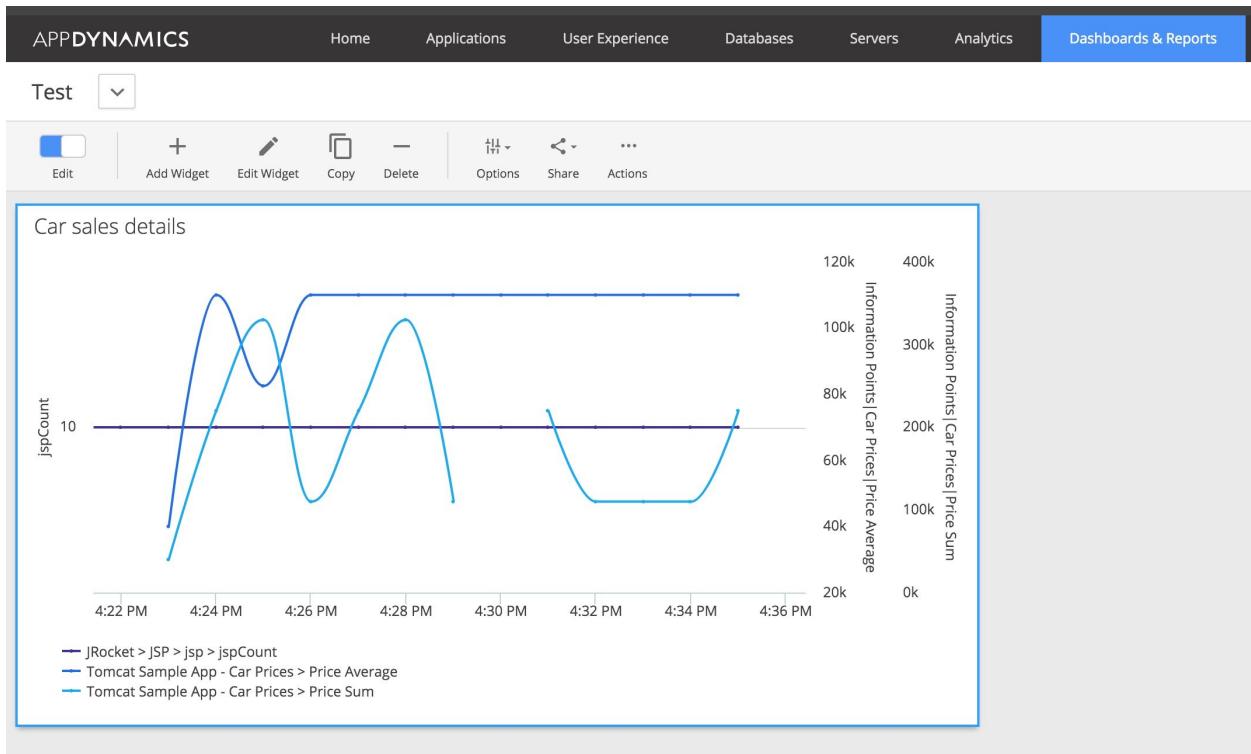
Custom Metric - Price Average

92000 average

Custom Metric - Price Average



Dashboard created by adding the custom metrics (JMX and Information Points).



Exercise 7: JMX Capture

Browse JMX through Applications/More/JMX.

Add a new metric through Configuration/Instrumentation/JMX. Create a new metric rule.

JMX Metric Rules

Tomcat JSP monitoring

Name	Enabled
Tomcat JSP monitoring	<input checked="" type="checkbox"/>

Tomcat JSP monitoring

Name: Tomcat JSP monitoring

Exclude Rule: Yes No

Enabled:

Metric Path: JSP

This is the path in the metric browser where this metric will be created relative to the JMX metric browser node.

And add the MBean config, Catalina:type=JspMonitor,WebModule=/localhost/Cars_Sample_App,name=jsp,J2EEApplication=none,J2EEServer=none with the jspCount, jspUnloadCount and jspReloadCount attributes.

MBeans

MBean Matching Criteria

Domain: Catalina

Object Name Match Pattern: Catalina:type=JspMonitor,WebModule=/localhost/Cars_Sample_App

Instance Identifier:

Advanced MBean Matching

Attributes

Define Metrics from MBean Attribute(s)

MBean Attribute	jspCount
Metric Name	jspCount

Advanced

MBean Attribute	jspUnloadCount
Metric Name	jspUnloadCount

Advanced

MBean Attribute	jspReloadCount
Metric Name	jspReloadCount

Advanced

Check the Metric Browser under Application Infrastructure Performance/TomcatAppServer/JMX for your metrics.

Exercise 8: Install EUM

Install EUM from Home/Getting Started Wizard/Real User. Set Browser Application.

2 Set Browser Application

Create a new Application:

You can always change this name later.

Tomcat

Continue

Click Continue and let the wizard complete (failed in my case each time).

Next link the created Browser Application with the Application already running through Applications/Configuration/User Experience App Integration by first enabling the Business Transaction Correlation.

User Experience App Integration

Business Transaction Correlation

JavaScript Agent Injection

Enable Business Transaction Correlation ⓘ

Next through JavaScript Agent Injection link the Browser Application created. Enable Automatic Injection of JavaScript. Save.

User Experience App Integration

Business Transaction Correlation

JavaScript Agent Injection

Inject the JavaScript Agent configured for this Browser App

Save

Automatic JavaScript Injection

Configure JavaScript Injection

Enable Automatic Injection of JavaScript ⓘ

Automatic injection must be enabled for specific Business Transactions. To enable a transaction: c
the list.

Manually generate load on the webpages and return to the JavaScript Agent Injection tab.
Move the discovered Business Transaction from right to left to enable Automatic Injection.

Automatic injection enabled (14)

Name	Tier
/	TomcatAppServer
/Cars_Sample_App/enquire.do	TomcatAppServer
/Cars_Sample_App/supercars.do	TomcatAppServer
/Cars_Sample_App/search.do	TomcatAppServer
/Cars_Sample_App/sell.do	TomcatAppServer
/Cars_Sample_App/insurance.do	TomcatAppServer
/Cars_Sample_App/about.do	TomcatAppServer

Automatic injection possible, but not enabled (0)

Name
< ADD >
REMOVE >

To check the EUMConfig in the log, tail -500 agent.2017_10_05__19_55_32.0.log | grep EUMConfig

```
[AD Thread Pool-Global4] 05 Oct 2017 20:38:45,691 INFO EUMConfig - EUM injected JS code template: ;(function(config){  
[AD Thread Pool-Global4] 05 Oct 2017 20:38:45,691 INFO EUMConfig - EUM cloud application key: AD-AAB-AAF-MHM  
[AD Thread Pool-Global4] 05 Oct 2017 20:38:45,691 INFO EUMConfig - Enabling automatic parsing and injection  
[AD Thread Pool-Global4] 05 Oct 2017 20:38:46,029 INFO EUMConfig - Enabling manual injection rules
```

Browse more pages and use the browser developers tools to check the Network traffic, the JavaScript should do a call to appdynamics.com (make sure to disable any plugins that block sending info).

	line.gif	java101template-can...	Image
	pipe.gif	java101template-can...	Image
	search_button.gif	java101template-can...	Image
	adrum-ext.84f248a459170522583669996382b...	cdn.appdynamics.com	Script
	adrum	col.eum-appdynamics...	XHR

View the instrumented page viewing the page source and see the code added add the top of the page.

All Resources Documents ▾

search.do — java101template-candidatejwjava...

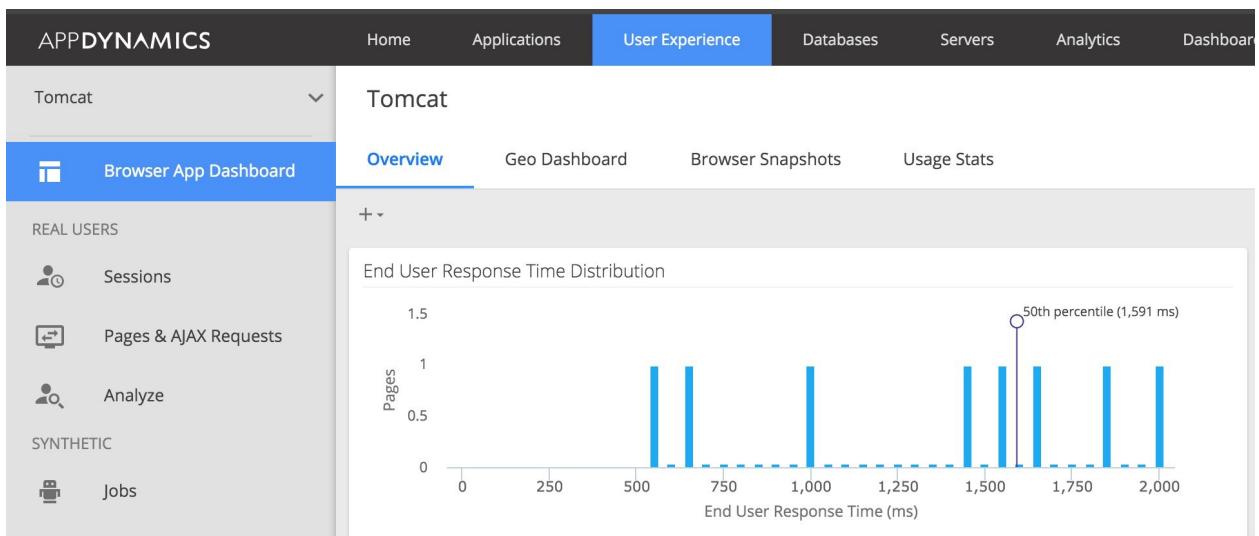
```

8 <html>
9   <head>
10  <script type='text/javascript'>
11  //<![CDATA[
12 window['adrum-start-time'] = new Date().getTime();
13 ;(function(config){
14 config.appKey = 'AD-AAB-AAF-MHM';
15 config.beaconUrlHttp = 'http://col.eum-appdynamics.com';
16 config.beaconUrlHttps = 'https://col.eum-appdynamics.com';
17 config.adrumExtUrlHttp = 'http://cdn.appdynamics.com';
18 config.adrumExtUrlHttps = 'https://cdn.appdynamics.com';
19 config.xd = {enable : false};
20 })(window['adrum-config'] || (window['adrum-config'] = {}));/* 
21 typeof a||!1}function d(a){return"string"==typeof a}function h(
22 b){return function(d,q){if(!f(d))throw new TypeError(a+" called
23 a"))}function +(a,c,d){return h.reduce(a,b,a,h)[c].call
</pre>

```

Once confirmed correctly setup start a new session in a Private Window.

Check the User Experience tab after some 15 minutes to see user traffic show up.



Session - 10/05/17 11:21:09 PM to 10/05/17 11:23:04 PM

Session Summary ▾			
Page Views in Session			
Page	Timeline	End User Response Time (ms)	
✓ * java101template-candidatejwjava101-lgflyzv1.srv.ravcloud.com/cars_sample_app/supercars.do	Start	2,024	
✓ * java101template-candidatejwjava101-lgflyzv1.srv.ravcloud.com/cars_sample_app/supercars.do	+3.650 sec	588	
✓ * java101template-candidatejwjava101-lgflyzv1.srv.ravcloud.com/cars_sample_app/supercars.do	+22.196 sec	1,591	
✓ java101template-candidatejwjava101-lgflyzv1.srv.ravcloud.com/cars_sample_app/search.do	+1 min 17.791 sec	1,473	
✓ java101template-candidatejwjava101-lgflyzv1.srv.ravcloud.com/cars_sample_app/search.do	+1 min 22.159 sec	1,851	
✓ * java101template-candidatejwjava101-lgflyzv1.srv.ravcloud.com/cars_sample_app/enquire.do	+1 min 43.747 sec	2,674	
✓ * java101template-candidatejwjava101-lgflyzv1.srv.ravcloud.com/cars_sample_app/supercars.do	+1 min 48.011 sec	1,000	
✓ * java101template-candidatejwjava101-lgflyzv1.srv.ravcloud.com/cars_sample_app/cars.do	+1 min 50.997 sec	677	
✓ * java101template-candidatejwjava101-lgflyzv1.srv.ravcloud.com/cars_sample_app/car.do	+1 min 53.869 sec	1,672	

Exercise 9: Machine Agent

Install Machine Agent from agent getting started, Home/Getting Started Wizard/Servers (download fails in my environment) or download machine agent from <https://download.appdynamics.com/>.

Agent installed and showing up but no data.

Tomcat Sample App									
OS	Name ↑	Hierarchy	Health	Disk Usage (%)	Disk Usage (%) Trend	CPU (%)	CPU (%) Trend	Memory (%)	Memory (%) Trend
- Java101Templa...	-	-	-	-	-	-	-	-	-

Exercise 10: Database Agent

Install DB Agent by selecting MySQL from the DB agent getting started, Home/Getting Started Wizard/Databases.

Download and transfer (FileZilla) zip file to DB server.

3 Configure the Controller

We've populated the connection information for this Controller for you. If you want to use another Controller, check your welcome email from AppDynamics for the host and port if you signed up for a SaaS account.

: 443 Use SSL

4 Download the Agent

It seems that a database agent is already running. You may proceed to step 6.

[Click Here to Download](#)

Unzip the Database Agent on a machine from which you can connect to the Controller and to the databases to be monitored.

Start the Database Agent:

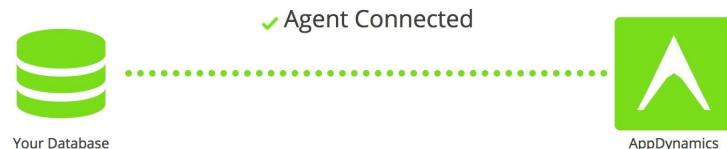
```
java -jar <db_agent_home>/db-agent.jar
```

<db_agent_home> is the location where the Database Agent was unzipped.

Extract file (/home/tomcat) and and start with OS user (tomcat).

Start agent: `java -Ddbagent.name='MySQLOnRavello' -jar /home/tomcat/dbagent/db-agent.jar &`

When 'Agent Connected' use form to fill out details and Continue.



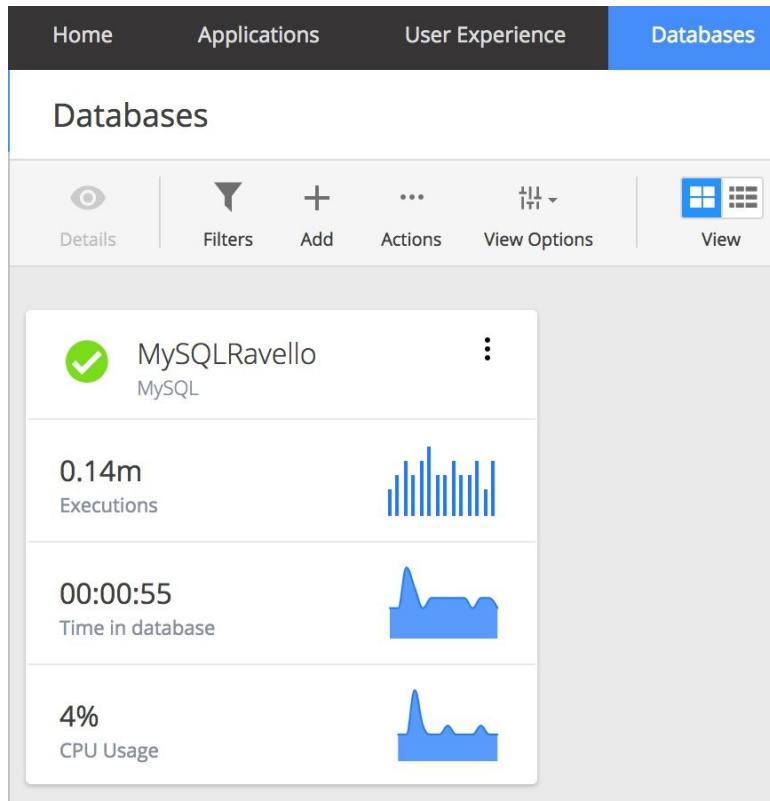
6 Configure the Database Collector

Required fields marked *

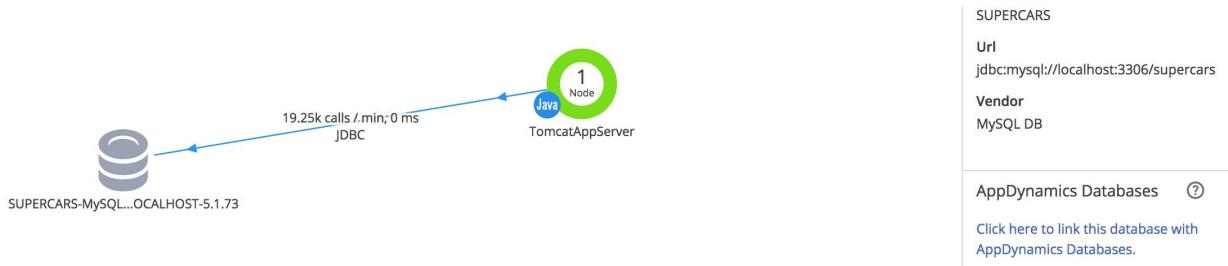
Name *	<input type="text" value="MYSQL"/>
Connection Details	
Hostname or Address *	<input type="text" value="127.0.0.1"/>
Listener port *	<input type="text" value="3306"/>
Username *	<input type="text" value="root"/>
Password	<input type="password" value="....."/>
or	
Custom JDBC Connection String <input type="text"/>	

[Continue](#)

Generate load and check the Databases tab.



On the Applications/Database Calls page link the DB and the application.



Select the configured DB and click OK.

Link SUPERCARS-MySQL DB-LOCALHOST-5.1.73

Select the database server to link this backend with. [?](#)

[Create New Collector](#)

Name	Hostname	Listener Port	Type
MySQLRavello	127.0.0.1	3306	MySQL

[Cancel](#) [OK](#)



[Dashboard](#) [Activity](#) [Live View](#) [Queries](#) [Clients](#) [Sessions](#) [Databases](#) [Object Browser](#)



SERVER HEALTH

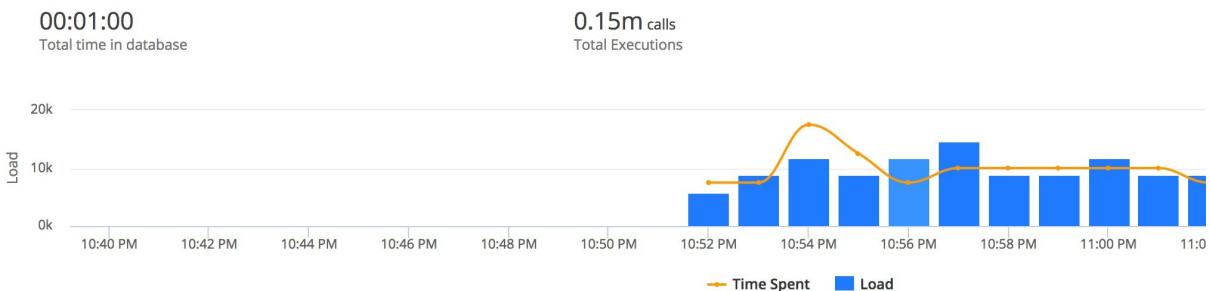


Normal

DATABASE TYPE

MySQL

Time Spent in Database and Executions



In Databases/Configuration also setup the hardware monitoring.

Hardware Monitoring

Monitor Operating System Operating System SSH Port Use certificate Enable CyberArk Username Password

CPU

0.9 %
System

2.1 %
User



Memory Used

34.2 %



Disk IO

43.0 KB/sec
Writes

0.0 KB/sec
Reads



Network IO

2.0 kB/sec
Outgoing

1.0 kB/sec
Incoming

