

# ArcGIS Server Performance and Scalability - Testing Methodologies

Andrew Sakowicz Frank Pizzi



# Introductions

## • Who are we?

-Enterprise Implementation

# • Who are you?

- -New to ArcGIS Server?
- -Currently deploying 9.3?
- -Currently deploying IMS?

# Audience

- Testers
- Developers
- Project Manager

# Level

# • Beginner / Intermediate

# Schedule and Overview





Performance and Scalability Testing methodology

 Web applications (step by step)
 ArcGIS Desktop Applications
 Exploratory tests

 Tips and Tricks

**Question and answer period.** 

Please complete the session survey!

### **Presentation Objectives**

# Teaching Step by Step testing (a simple web Flex API)

- -Recording and creating transactions scripts
- -Creating load test
- -Executing
- -Analyzing Results

# Managing testing efforts

- -Test Plan, skills, level of effort
- -Best practices
- -Common pitfalls

# **Demo** Example of ArcGIS Server Flex application used for parcel management by local government

Flex application with dynamic data. Application provides the search and query functions as well as Geoprocessing



# Demo Define user workflow, transactions and think time

- Define user workflow to be tested
- Define and name GIS transactions
  - Initial page of application
  - Zoom map 1
  - Zoom map 2
  - Pan map 1
  - Pan map 2
  - -Query tax lot with zoom to tax lot
  - -Query school with zoom to school
  - -Query hospital with zoom to
  - Geoprocessing
- Define user think time between transactions

# **Demo – Testing tasks**

Record user workflow based on application user requirements

- Create single user web test
  - -Define transactions
  - Set think time and pacing based on application user requirements
  - -Parameterize transaction inputs
  - -Verify test script with single user
- Create load test
  - -Define user load
  - -Create machine counters to gather raw data for analysis
  - -Explain variable load test variables
- Execute

# Demo

# What is measured during a typical web load test

Performance measured at web server tier, not a web browser – not a true user experience



# How to measure true web user performance experience?

Measure at web browser, e.g. stop watch or programmatically using API events



### How to measure true web user performance experience?

Measure at web browser, e.g. stop watch or programmatically using API events



### How to measure true web user performance experience? Measuring map rendering at web browser using API events

```
private function onCreationComplete():void
{
    logger.text += getTimer() + ' ms: Application creation completed.\n';
    myThematicLayer.addEventListener(Event.COMPLETE, onTilesLoaded)
    myMap.addEventListener(PanEvent.PAN_START, onPanStart);
    myMap.addEventListener(ZoomEvent.ZOOM START, onZoomStart);
```

```
private function onTilesLoaded(event:Event):void
```

}

{

}

```
if (panStartFlag){
    TimeTileComplete = getTimer() - TimeTileStart ;
    TimeTileStart = 0
    logger.text += "Tiles took " + TimeTileComplete + " ms to pan.\n";
    panStartFlag = false;
}
if (zoomStartFlag){
    TimeTileComplete = getTimer() - TimeTileStart ;
    TimeTileStart = 0
    logger.text += "Tiles took " + TimeTileComplete + " ms to Zoom.\n";
    zoomStartFlag= false;
}
```

### How to measure true web user performance experience? Measuring client-side graphics at web browser using API events

```
private function doThematicQuery():void
{
    myGraphicsLayer.clear();
    queryTask.url = textQueryURL.text.toString();
    logger.text += getTimer() + " ms: Starting query...\n";
    TimeQueryStart = getTimer();
    queryTask.execute( query, new AsyncResponder( onResult, onFault ));
    myGraphicsLayer.addEventListener(FlexEvent.UPDATE_COMPLETE,graphicsLayerUpdated);
    function onResult( featureSet : FeatureSet, token : Object = null ) : void
    {
        TimeQueryDone = getTimer();
        var TimeQueryDiff:uint = getTimer() - TimeQueryStart;
        logger.text += getTimer() + " ms: Query took " + TimeQueryDiff + " ms";
        logger.text += " [" + myGraphicsLayer.numGraphics + " features].\n";
    }
```

### Resources

http://resources.esri.com/help/9.3/arcgisserver/apis/flex/samples/index.html

- The ESRI Flex and JavaScript APIs are straight forward to test
- Flex and JavaScript APIs utilize the REST API to access GIS data
- Understanding the REST API Interface will provide insight in how to approach testing
   ArcGIS Resource Centers
  - Map Service
  - Geocode Service
  - GP Service
  - Geometry Service
  - Image Service



Good resource to practice test scripting

# Managing testing efforts

# **Performance Testing Objectives**

 Benchmarking to demonstrate performance improvements

- Part of formal acceptance testing
- Capacity planning (server sizing)
- Validating system architecture design

# **Performance Testing Benefits**

- Less costly to fix the problems at the earlier development stage
- Proactive resolution of issues before they impact on end users
- Identify bottlenecks and/or underutilized resources for redeployment

**Performance Testing Methodology** 

Ensure application functional testing is complete

Tune your system prior to testing

Develop test plan

Create test scripts

Analyze results

# Performance Testing Methodology – Tuning Analyze performance of each request

🕑 Web Mapping A	pplication - Mozilla Firefox							
Ele Edit Yew H	igtory Bookmarks Iools Help			0				
🦛 • 🧼 • 🎯	🛞 🕋 🕒 http://localhost/lidtPooled/	<b>G</b> • <b>c</b>	ogle	<b>Q</b> .)				
Web Mapping	g Application	-	K					
o o 🤭 🚱	💽 👑 🔂 👘 👘 🖏	er sopp	ort Centr	er i stelp				
Tasks	Request Graph - Mozilla Firefox	_						EE
Editing	Elle Edit View History Bookmarks Tools Help							
	http://localhost/EditPooled/Default.aspx	200	892					
	http://localhost/EditPooled/ESRLArcGIS ADF Web Mimelmane astro?	200	410					
e e ultre	http://localhost/EditPooled/Default.aspx	200	1553					
esolts	http://localhost/EditPooled/Default.aspx	200	862	<b>i</b>				
ap Contents	http://localhost/EditPooled/ESRLArcGIS ADF Web Mimelmage astro?	200	30	- T				
avigation	http://localhost/EditPooled/Default.aspx	200	501					
verview	http://localhost/EditPooled/ESRLArcGIS.ADF.Web.Mimelmage.ash/?	200	20		1			
	http://localhost.EditPooled.Default.aspx	200	5749		The second s			
	http://localbost/EditPooled/Default.aspx	200	5048					
	http://iocathosbEditPooled/ESRLArcGIS ADF Web Mimelmage astw?	200	90		-			
	http://localhost.EditPooled.Default.aspx	200	19399			1		
SAN DIEGO BAY	http://localhost.EditPooled.Default.aspx	200	18458			-		
	http://localhost.EditPooled/Default.aspx	200	17737					
	http://iocalhost/EditPooled/Default.aspx	200	17236					
SAN DIEGO BAY	http://localhost.EditPooled.Default.aspx	200	13149					10 M
	http://iocalhost.#ditPooled/Default.aspx	200	12679					
ne	http://localhost.EditPooled/Default.aspx	200	5759			11		
	http://localhost/EditPooled/ESRLArcGIS ADF Web Mimelmage ashr?	200	11858					100 C
	http://localhosbEditPooled/ESRLArcOIS.ADF.Web.Mimelmage.ash/?,	200	5128					
	http://sb.google.com/safebrowsing/update?	200	110					1
	http://localhost/EditPooled/ESRI.ArcGIS.ADF.Web.Mimelmage.ashx?	200	1072					
	http://localhosbEditPooled/ESRI.ArcGIS.ADF.Web.Mimelmage.asht?	200	30					T
	URI	Status	Duration	 15:26:00.086	 15:26:09.028	15:26:17.970	 15:26:26.911	15:26:35.853

### Performance Testing Methodology – Tuning example Optimize mxd with new ArcMap 9.3.1 Analyze Tool

### Rectional Content of the second secon

	File Edit View Bookmarks Insert Selection Tools Window Help									
]	🗋 😂 🔚 🎒 👗 🖻 🛍 🗙 🗠 🗠 🔸 [1:8,976									
Portland   Places   Places   Schools   Cityhall   Cityhall   Fire Stations   Hospitals   Libraries   Airports   Transportation										
<b>X</b>	8	) Errors	🔥 6 Warnin	gs 🛛 🤅	2 Messages Search:					
		Severity	Status	Code	Description					
		Medium	Unresolved	20007	Layer contains a multilayer line symbol whose					
		Medium	Unresolved	10009	Enabling the option to convert layer transpare					
		Medium	Unresolved	10009	Enabling the option to convert layer transpare					
	<u> </u>	Medium	Unresolved	10010	Raster layer's data source does not have imag					
	<									

# **Performance Testing Methodology – Tuning example**

### Optimize mxd with mxdperfstat

41

42

1,000 Taxlot Dense Label

1,000 TaxlotDenseNoLabel

http://arcscripts.esri.com/scripts.asp?eLang=&eProd=&perPage=10&eQuery=mxdperfstat

polyline: vertices fetched=200001; simplify geometry and/or set label scale: vertices

1.93 fetched=200001;

0.53 fetched=200001;

simplify geometry: vertices

		Strendpectatet Anti-regive Appl	tication:									
		The rest of the re							Contraction of the			
			9 🗰 #8  124.000	三 1 4 4 0 益								
		Portland     K      Know     Konset     Konset     K      Know     Konset     Konset     K     Konset		A market	B	and and a second se						
ltem	At Scale	Layer Name	Refresh Time (sec)	Recommendations	Features	Vertices	Labeling	Geography Phase (sec)	Graphics Phase (sec)	Cursor Phase (sec)	DBMS CPU	DBM LIO
				Simplify labeling, symbology: GraphicsPhase=1.42; simplify geometry and/or set label scale; convert polygon to								

200,001

200,001

TRUE

FALSE

0.45

0.45

1.42

0.02

1.04

0.9

266

140

0.02

0.02

### Performance Testing Methodology – Tuning example Optimize ArcMap document and configure correct number of ArcGIS Server instances



Corrected (Unloaded TT: 0.34 sec) (2.1 Instances/core)

Uncorrected (Unloaded TT: 11.97 sec) (1.6 Instances/core)

# When?

# Start performance testing early in the development cycle Unit test

# **Test Plan**

 Understand the effort required to develop and ensure testing is valid:

- -Domain experts
- -Hardware
- -Time
- -Skills
- Document user workflows
- Document configuration

# **Selecting Web Load Test Tool**

Test Tools	Open Source	Pros	Cons
LoadRunner No •Industry Lead •Automatic ne with service le •Http Web Te •Click and Sc •Very good to •Test results •Thick Client		<ul> <li>Industry Leader</li> <li>Automatic negative correlations identified with service level agreements</li> <li>Http Web Testing</li> <li>Click and Script</li> <li>Very good tools for testing SOA</li> <li>Test results stored in database</li> <li>Thick Client Testing</li> <li>Can be used for bottleneck analysis</li> </ul>	<ul> <li>High Cost</li> <li>Test Development in in C programming language</li> <li>Test metrics difficult to manage and correlate</li> <li>Poor user community with few available examples</li> </ul>
Silk Performer	No	<ul> <li>Good solution for testing Citrix</li> <li>Wizard driven interface guides the user</li> <li>Can be used for bottleneck analysis</li> </ul>	<ul> <li>Moderate to High Cost</li> <li>Test metrics are poor</li> <li>Test Development uses proprietary language</li> <li>Test metrics difficult to manage and correlate</li> <li>Poor user community with few available examples</li> </ul>
Visual Studio Test Team	No	<ul> <li>Low to moderate cost</li> <li>Excellent Test Metric reporting</li> <li>Test Scripting in C# or VB .NET</li> <li>Unit and Web Testing available</li> <li>Blog support with good examples</li> <li>Very good for bottleneck analysis</li> </ul>	<ul> <li>No built in support for AMF</li> <li>No Thick Client options</li> <li>Moderate user community</li> </ul>
JMeter	Yes	•Free •Tool	<ul> <li>Provides only response times</li> <li>Poor User community with few available examples</li> </ul>

# Execute

# Ensure

- -Only target applications are running
- -Application data is in the same state for every test
- Good configuration management is critical to getting consistent load test results

### Analyze Results Expected Response Time Curve



# Analyze Results Validation

# Compare and correlate key measurements

- Response Time (increasing, higher than initially profiled for single user)
- -Throughput
- -CPU on all tiers
- -Network on all tiers
- -Disk on all tiers
- -Passed tests
- -Failed test



### Lack of errors does not validate a test

- -Requests may succeed but return zero size image
- -Spot check request response content size



 Exclude failure range, e.g. failure rate > 5% from the analysis

# Exclude excessive resource utilization range

### Analyze Results Validation Example



### Analyze Results Expected results example



### Analyze Results Failing test case



### Analyze Results Failing test case


### **Analyze Results**

Determining system capacity

### Maximum number of concurrent users corresponding to:

- -Maximum acceptable response time
- -First failure or 5%
- -resource utilization greater than 85%, for example CPU

### Different projects define acceptance criteria (performance level of service), e.g.

- -95% of requests under 3 sec
- -Max request under 10 sec

### **Common pitfalls**

- Application stack is not properly configured and tuned
- Single error in a test can lead to a cascade of failing requests

### Test client machine is bottleneck

- -CPU
- -Memory
- -Networks (cache test in particular)

### **Common pitfalls – configuration problem test case**



Read

## **Demo – Analyzing Test Case Results**

### ESRI Flex, JavaScript, and REST Resources

### • Flex API

- <u>http://resources.esri.com/help/9.3/arcgisserver/apis/flex/samples/index</u>
   <u>.html</u>
- JavaScript API
  - <u>http://resources.esri.com/help/9.3/arcgisserver/apis/javascript/arcgis/help/jssamples\_start.htm</u>
- REST API

- http://resources.esri.com/help/9.3/arcgisserver/apis/REST/index.html

### References

#### Fiddler

- http://www.fiddlertool.com/fiddler/version.asp

#### Visual Studio 2008 Professional Edition (90-day Trial)

<u>http://www.microsoft.com/downloads/details.aspx?familyid=83C3A1EC-ED72-4A79-8961-25635DB0192B&displaylang=en</u>

#### MSDN

- Chapter 17 Load-Testing Web Applications
  - http://msdn.microsoft.com/en-us/library/bb924372.aspx
- Creating a Web Test
  - http://msdn.microsoft.com/en-us/library/ms182538.aspx

#### • BLOGS

- Bill Barnett's blog
  - <u>http://blogs.msdn.com/billbar/pages/load-test-api-enhancements-in-vsts-2008-sp1-beta.aspx</u>
- Ed Glas's blog on VSTS load testing
  - http://blogs.msdn.com/edglas/archive/2006/02/06/525614.aspx
- Sean Lumley's Blog
- http://blogs.msdn.com/slumley/archive/2008/12/22/modifying-request-properties-with-aweb-test-plugin.aspx

**Performance Testing ArcMap and ArcEngine Applications** 

Several tools can be used to automate thick client workflows
–HP LoadRunner
–Borland's Silk Perfomer
–NRG's AppLoader

Better choices are text indicators that eventually have a definitive ending (such as text stating "complete", or no text at all)



Measuring pan and zoom transactions

### • WaitBitmapAppear:

- lower left status text continuously update when drawing occurs, and will become blank when ended
- Testing software waits until the bitmap is matched before proceeding.
- Specify a search area when capturing the bitmap.
- Keep a search area small
- Animated icons or other moving status indicators are particularly difficult to implement (typical example is the spinning globe in ArcMap)
- Better choices are text indicators that eventually have a definitive ending (such as text stating "complete", or no text at all)

Measuring pan and zoom transactions (Apploader tool example)

롣 Capture Dit	map 🔀	
Parameters-		
Width:		
Height:	-j	
Threshold:		
Name:	Bitmap1236681840	
Wait Time:	10	
Search Area:	0,0,0,0 Specify	
Cursor Postic	n	
Show Curse	or Position Grid	
ОК	Cancel Reset	

Measure opening ArcMap document (Apploader tool example)

### Wait for Window

#### Select Window



Select the window by either using the drag icon or from the list

#### Window List

Clear buffer cache and shared pool in Orac Document1 - Microsoft Word eslsrv17 - Remote Desktop Machine Sheet - Windows Internet Explore PortlandFGDB.mxd - ArcMap - ArcInfo Remote Desktop Connection

Window Title

PortlandFGDB.mxd - ArcMap - ArcInfo

Window Class

Afx:00400000:b:00010011:0000006:030

Wait Time

30

0k

Cancel

### Capturing Transactions in ArcMap Using Events Open ArcMap, Zoom, Close ArcMap

- 1. Comment : Open ArcMap
- 2. Launching ArcMap via specified MXD
- 3. Waiting for the ArcMap window to appear
- 4. Waiting for captured bitmap to appear. This bitmap is a blank status area indicating all drawing has completed.
- 5. Comment: Zoom
- 6. Move the mouse to the map scale field
- 7. Click the map scale field
- 8. Type the scale we wish to zoom to
- 9. Hit enter, initiating the map zoom
- **10.** Comment: Start Zoom Transaction
- **11.** Changing the Event Interval to 50ms for better transaction capturing
- **12.** Begin the transaction
- **13.** Wait for bitmap to appear. This bitmap is the grey status area at the lower left of the screen. It will become blank when the map has finished drawing
- 14. End the transaction
- **15.** Changing the Event Interval back to its default 1000ms
- **16.**Comment: Close ArcMap
- **17.** Closing ArcMap via shortcut

Applorater Matters - Constant Loss	19.65	
the task tax tele		
DOB B PPERON	II AYX DIP	
Actionsi C Laurch Program C Laurch Program Per Stemato Workers Workers Workers Workers Workers Restor Worker Restor Restor Restor Restor Restor Restor Restor Restor Restor Restor Restor Restor Restor Restor Restor Restor Restor Restor Restor	Lee Connect Doct Antitian van mit Connect Doct Antitan van mit Lee Connect Doct Antitan van mit Varbinske Start (Connect Connect Conneconnect Connect Connect Connect Connect Connect Connect Connect C	24 mail and Anthon Anthon Antonia Control to
Converted to AppLander Canada		2

## Exploratory tests

### Using synthetic data for exploratory tests

### Benefits

- -Controlled density
- -Easy to script

	Lord St. Ala head
E Di Santa hann E Di Santa hann E Di Santa hann E Di Santa (Janager,	
2000 <del>-0</del> 12-020-7	

# Generating synthetic data RandomEdits -a custom gp tool available for download

Input Feature Class (optional)	<u>م</u>	RandomEdits
E Has AssSDE connection assumption	2	Generate Random Edits in
i use Arcabic connection parameters		a FeatureClass
Server (optional)	-	
And any a Continue (		
Database (optional)	- 10	
Jser (optional)		
(che)		
Password (optional)		
(ersion (optional)	-	
Feature (internal)		
sole Tadting	18	
Number of features to be generated		
10	0	
/ertices per feature + Min		
	3	
/ertices per feature - Max		
	2	
reature envolope length - min	0	
Fash as equiphes length - may		
Const & Contransport Periodicia - Interna	0	
Area of interest: Xmin		
	100	
Area of interest: Ymin		
	- 21	
Area of interest: Xmax	-	
A CONTRACTOR OF		
Area of interest: Ymax	-	
	1	
	>	

### Using synthetic data for exploratory tests

Helps analyzing generic performance factors, e.g. Fetch time vs. # features



### Summary

### • Today we covered:

### Performance and Scalability Testing methodology

- -Web applications (step by step)
- **–ArcGIS Desktop Applications**
- -Exploratory tests
- Tips and Tricks

Still have questions?

# Additional Resources

Questions, answers and information...

Tech Talk

Outside this room right now!

Other sessions

 ArcGIS Server Performance and Scalability – Performance Factors  ESRI Resource Centers
 DBTs, code and video



PPTs, code and video resources.esri.com



facebook

tinyurl.com/ ESRIDevSummitFB

### **Questions?**

Frank Pizzi, <u>fpizzi@esri.com</u> Andrew Sakowicz, <u>asakowicz@esri.com</u> ESRI Professional Services requests: John Graham, jgraham@esri.com Chad Helm, <u>chelm@esri.com</u>

## **Demo Related and Supplemental Slides**

### **Using Visual Studio for web testing**

### Creating web test script

- **1. VS Recorder** 
  - might not detect all parameters for ADF
  - must validate with Fiddler or other tools
  - may have to edit registry entries
    - -http://msdn.microsoft.com/en-us/library/cc678655.aspx
- **2.** Fiddler and export to VS Web test
  - Deselect appropriate plug-in
- 3. Automated solutions (especially useful for ADF testing), e.g. Fiddler custom plug-in
- Declarative vs. Code web test
  - Looping and if logic can be implemented only in coded web test

### Scripting Stateless vs. Stateful applications

### Examples

-Stateless: JavaScript, Flex, SilverLight API

-Stateful: .NET and Java ADF

### State management

- -Stateful
  - Stores sessions in a web server process (InProc)
  - Often web server memory bound
- -Stateless
  - Manages state in a browser

Challenges of testing

-Stateful application more difficult to test

### **Scripting applications - Challenges**

### Stateful application

- -Test scripts must manage the state
- -Most requests contain session based or dynamic GUID
  - Map Images
  - Scale bar
  - North Arrow
  - Search task

Parameterize request

### **Creating Transaction Scripts – manual approach**

- Most load testing tools include a an HTTP debugging proxy to model user workflows. The output from the HTTP Debugging proxy is raw text or another format that represents the Web Client requests.
- When recording user workflows, use transaction markers between the transactions. For Example, initial page load, zoom, and pan are transactions
- The transaction marker is also the correct place to insert think time as a user would naturally pause after a transaction

### **Creating Transaction Scripts - manual approach**

	84	200	HTTP	eslsrv11	/TemplateMixedAggregate/WebResource.axd?d=InwIAjzTW5r2YzMrDnvYp-t8uIToH6LbDqCa-F
≣	85	200	HTTP	eslsrv11	/TemplateMixedAggregate/WebResource.axd?d=InwIAjzTW5r2YzMrDnvYp-t8uIToH6LbDqCa-F
≣	86	200	HTTP	eslsrv11	/TemplateMixedAggregate/WebResource.axd?d=InwIAjzTW5r2YzMrDnvYp-t8uIToH6LbDqCa-F
•	87	200	HTTP	eslsrv	/TemplateMixedAggregate/ESRI.ArcGIS.ADF.Web.MimeImage.ashx?ImgID=d20
≣	88	200	HTTP	eslsrv11	/TemplateMixedAggregate/WebResource.axd?d=InwIAjzTW5r2YzMrDnvYp-t8uIToH6LbDqCa-F
•	89	200	нттр	eslsrv	/TemplateMixedAggregate/ESRI.ArcGIS.ADF.Web.MimeImage.ashx?ImgID=c580
2	90	200	HTTP	eslsrv11	/TemplateMixedAggregate/images/dismiss.png
2	91	200	HTTP	eslsrv	/TemplateMixedAggregate/ESRI.ArcGI5.ADF.Web.MimeImage.ashx?ImgID=5d5<
A	92	502	HTTP	INI	
2	93	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L06/R0000bda3/C0000a0e5.jpg
2	94	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L06/R0000bda4/C0000a0e5.jpg
≣	95	200	HTTP	eslsrv11	/TemplateMixedAggregate/ESRI.ArcGIS.ADF.Web.UI.WebControls.MapHandler.ashx?MapID=f
	96	200	HTTP	eslsrv11	/TemplateMixedAggregate/Default.aspx
	97	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L06/R0000bda3/C0000a0e6.jpg
	98	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L06/R0000bda4/C0000a0e6.jpg
	99	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L06/R0000bda3/C0000a0e7.jpg
	100	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L06/R0000bda4/C0000a0e7.jpg
	101	200	HTTP	eslsrv11	/TemplateMixedAggregate/images/backward.png
	102	200	нттр	eslsrv	/TemplateMixedAggregate/ESRI.ArcGIS.ADF.Web.MimeImage.ashx?ImgID=b2e <sup>-</sup>
2	103	200	нттр	eslsrv	/TemplateMixedAggregate/ESRI.ArcGIS.ADF.Web.MimeImage.ashx?ImgID=305
A	104	502	HTTP	Z1	
2	105	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L07/R0001602f/C00012acf.jpg
2	106	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L07/R00016030/C00012acf.jpg
2	107	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L07/R00016031/C00012acf.jpg
	108	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L07/R0001602f/C00012ad0.jpg
	109	200	HTTP	eslsrv11	/TemplateMixedAggregate/Default.aspx
	110	200	HTTP	eslsrv11	/TemplateMixedAggregate/ESRI.ArcGIS.ADF.Web.UI.WebControls.MapHandler.ashx?MapID=f
	111	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L07/R00016030/C00012ad0.jpg
	112	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L07/R00016031/C00012ad0.jpg
	113	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L07/R0001602f/C00012ad1.jpg
	114	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L07/R00016030/C00012ad1.jpg
	115	200	HTTP	eslsrv11	/arcgiscache/PortlandCacheNoTaxLots/Portland/_alllayers/L07/R00016031/C00012ad1.jpg
	116	200	HTTP	esisry	/TemplateMixedAggregate/ESRI.ArcGIS.ADF.Web.MimeImage.ashx?ImgID=210
2	117	200	HTTP	esisry	/TemplateMixedAggregate/ESRI.ArcGIS.ADF.Web.MimeImage.ashx?ImgID=c65t
A	118	502	HTTP	Z2	1

### **Creating Transaction Scripts – manual approach**

- Using the transaction markers we will define GIS transactions.
- The content returned from the initial page request will contain dynamic images that must be captured, formatted properly, and sent back to the server in the form of an HTTP GET.
- All references to HTTP GET with ESRI.ArcGIS.Web.MimeImage.ashx must be altered to reflect the new session
- Transaction markers must be removed

### **Creating Transaction Scripts – Fiddler manual**

- Using Fiddler, an HTTP Debugging proxy, record the desired Web client workflow
- Be sure to include the transaction markers in your workflow.
- When you have completed recording the defined workflow, select all (Ctrl-A) and go to File->Save->Session(s)->as Visual Studio Web Test and save the raw HTTP traffic to fiddler.webtest. You will be prompted to select VS2008 plugins on export. Please select the following options-

Select Plugins	
<ul> <li>Fiddler.WebTesting.ParseDependents</li> <li>Fiddler.WebTesting.CorrelateEventValidation</li> <li>Fiddler.WebTesting.FilterByMimeType</li> <li>Fiddler.WebTesting.FilterBetrects</li> <li>Fiddler.WebTesting.FilterAuthRequests</li> <li>Fiddler.WebTesting.CorrelateViewState</li> </ul>	
	OK Cancel

### **Creating Transaction Scripts – VS Web Test**

🗄 🖳 🌺 http://esisrv2/Template5050/

### Declarative webtest

🗄 🕞 http://eslsrv2/Template5050/App_Themes/Blue_Bridge/images/TitleBanner.jpg
🗄 🕞 http://eslsrv2/Template5050/App_Themes/Blue_Bridge/images/blue-gradient-30x1.gif
🗄 🕞 http://eslsrv2/Template5050/App_Themes/Blue_Bridge/images/blue-gradient-24x1.gif
🗄 💽 http://eslsrv2/Template5050/North_Arrow/directional_arrows_N.gif
🗄 🖳 http://eslsrv2/Template5050/images/dismiss.png
🗄 🖳 http://esisrv2/Template5050/WebResource.axd
🗄 🕞 http://eslsrv2/Template5050/WebResource.axd
🗄 🖳 http://eslsrv2/Template5050/WebResource.axd
🗄 🖳 http://eslsrv2/Template5050/WebResource.axd
🗄 🖳 http://eslsrv2/Template5050/ESRI.ArcGIS.ADF.Web.MimeImage.ashx
🗄 🕞 http://eslsrv2/arcgiscache/PortlandCache/Portland/_alllayers/L00/R000003d9/C00000343.png
🗄 🕞 http://eslsrv2/arcgiscache/PortlandCache/Portland/_alllayers/L00/R000003da/C00000343.png
🗄 🕞 http://eslsrv2/arcgiscache/PortlandCache/Portland/_alllayers/L00/R000003d9/C00000344.png
🗄 🕞 http://eslsrv2/arcgiscache/PortlandCache/Portland/_alllayers/L00/R000003da/C00000344.png
🗄 🕞 http://eslsrv2/arcgiscache/PortlandCache/Portland/_alllayers/L00/R000003d9/C00000345.png
🗄 🕞 http://eslsrv2/arcgiscache/PortlandCache/Portland/_alllayers/L00/R000003da/C00000345.png
🗄 🕞 http://esisrv2/Template5050/ESRI.ArcGIS.ADF.Web.UI.WebControls.MapHandler.ashx
🗄 🕞 http://esisrv2/Template5050/WebResource.axd
🗄 🕞 http://esisrv2/Template5050/WebResource.axd
🖶 🕞 http://esisrv2/Template5050/WebResource.axd
🖶 🕞 http://esisrv2/Template5050/WebResource.axd
连 🕞 http://esisrv2/Template5050/ESRI.ArcGIS.ADF.Web.MimeImage.ashx
连 🕞 http://esisrv2/Template5050/WebResource.axd
🖶 🕞 http://eslsrv2/Template5050/ESRI.ArcGIS.ADF.Web.MimeImage.ashx
💁 http://myserverTM-/think.html
🖶 🕞 http://eslsrv2/arcgiscache/PortlandCache/Portland/_alllayers/L01/R00000ab7/C00000917.png
连 🕞 http://eslsrv2/arcgiscache/PortlandCache/Portland/_alllayers/L01/R00000ab8/C00000917.png
连 🕞 http://eslsrv2/arcgiscache/PortlandCache/Portland/_alllayers/L01/R00000ab7/C00000918.png
连 🕞 http://eslsrv2/arcgiscache/PortlandCache/Portland/_alllayers/L01/R00000ab8/C00000918.png
连 🕞 http://eslsrv2/Template5050/ESRI.ArcGIS.ADF.Web.UI.WebControls.MapHandler.ashx
连 🖳 http://esisrv2/Template5050/Default.aspx
🗄 🛃 http://eslsrv2/Template5050/images/backward.png
🖹 🗠 🚰 http://esisrv2/Template5050/ESRI.ArcGIS.ADF.Web.MimeImage.ashx
🗄 🛅 QueryString Parameters
🗄 💼 Headers

### **Creating Transaction Scripts – VS Web Test manual**

### Group requests into transactions

Transaction name:		
Transaction1		
Choose first item for transaction:		
http://esisrv2/Template5050/		<b>~</b>
Choose last item for transaction:		
http://myserverTM-/think.html		<b>•</b>
	ок	Cancel

### Define extraction rules for each web request that returns dynamic images.

 Extraction rules define search patterns used to parse web requests, and the results of the search are held in a variable defined in extraction rule.

# Creating Transaction Scripts – VS Web Test manual extraction Rules

Add Extraction Rule		<u>?×</u>				
Select a rule: Extract Attribute Value Extract Form Field Extract HTTP Header Extract Regular Expression Extract Text Extract Hidden Fields	Properties for selected rule:	False True 0				
Description for selected rule: Extract text from the response matching a regular expression and place it into the test context. OK Cancel						

### Creating Transaction Scripts – VS Web Test manual extraction rules for ADF testing

Parameters	Value	Notes
Context Parameter Name	t0-i0	This item represents a variable that will be set when the extraction rule is executed on the response of an HTTP request. I recommend following the convention to the left. T0 represents transaction0 and i0 represent image0. If there is more than one request coming back in the response of the server or the define transaction has multiple dynamic images the Context Parameter name will simply be advanced. For Example, t0-i2.
Regular Expression	ImgID=((([0-9a-fA-F]){8}-([0-9a-fA-F]){4}-([0-9a-fA- F]){4}-([0-9a-fA-F]){4}-([0-9a-fA-F]){12}) ([0- 9a-fA-F]){32})	Dynamic images will have this signature, and the regular expression will find multiple images if they are present in the HTTP response from the server
Ignore Case	False	
Required	True	
Index	0	The regular expression will return an array of item that matches the search criteria. This number represents the index of the array that must be dereferenced.

### Creating Transaction Scripts – VS Web Test manual extraction rules for ADF testing

- Now, we have defined all of the extraction rules for the first transaction.
- Next, we must find the static requests from the initial HTTP Debugging proxy and modify them to be dynamic.
- Below is an example of a static request for a mime image that must be modified.
- We will delete all of the query string parameters, but we must leave the headers alone.



# Creating Transaction Scripts – VS Web Test manual extraction rules for ADF testing

- Now, we will use the variable defined by the Context Parameter Name in the extraction rules
- As you can see, we added the {{t0-i0}} to the correct request parameter. This modification to the request will dereference the variable that has been set when the extraction rule is executed.

**Modeling Web Clients** 



# **Creating load test – Visual Studio**

New Load Test Wizard		<u>? ×</u>
Edit load patter	n settings for a load test scenario	
Welcome Scenario Load Pattern Test Mix Browser Mix Network Mix Counter Sets Run Settings	Select a load pattern for your smullated load:         Constant Load:         User Count:       25          User Count:       25          Step load:         Start user count:       1          120        seconds         Step user count:       1          Maximum user count:       1          4        users	
	< Previous Next > Finish Cancel	

# **Creating load test – Visual Studio**

New Load Test Wizard						? ×
Add tests to a load test scenario and edit the test mix						
Welcome	Add	one or more tests to the mix and specify a	distributio	n:		
Scenario		Test Name	%	Distribution	۵	Add
Load Pattern	1	PortlandDynamicTemplate-Final	100	· · · · · · · · · · · · · · · · · · ·		Remove
Test Mix						Distribute
: Browser Mix						
Network Mix						
Counter Sets						
Run Settings						
		Total	100			
		< Prev	ious	Next > Finish	٦	Cancel

# **Creating load test – Visual Studio**

New Load Test Wizard							
Add network types to a load test scenario and edit the network mix							
Welcome	Add one or more network types to the mix and specify a distribution:						
Scenario		Network Type	%	Distribution	٩	Add	
Load Pattern	1	LAN	100	· · · · · · · · · · · · · · · · · · ·	Г	Remove	
Test Mix						Distribute	
: Browser Mix						Discribuce	
Network Mix							
Counter Sets							
• Run Settings							
		Total	100				
	< Previous Next > Finish Cancel						
# **Creating load test – Visual Studio**

New Load Test Wizard		? ×	
Specify computers to monitor with counter sets during load test run			
Welcome Scenario	Selected computers and counter sets will be added to to Computers and counter sets to monitor:	he default run settings Preview selections:	
Load Pattern Test Mix	ADO.Net	Application	
Network Mix Counter Sets	Net Application	IIS E-S_Controller Computer	
Run Settings		Agent	
	Add Computer Remove		
	< Previous	Next > Finish Cancel	

### **Running a Load Test with System Monitoring**

While the load test is running, various system components should be monitored including the CPUs, disk subsystem, memory, and network. System monitoring applies to all the systems that play a part in the test configuration, including the database server. At a minimum, the following core components should be monitored during the tests:

- CPU Utilization
- Memory
- Disk Utilization
- Network Utilization

# **Creating load test – Visual Studio**

New Load Test Wizard	<u>?</u>	×		
Review and edit run settings for a load test				
Welcome Scenario Load Pattern Test Mix Browser Mix	Timing   Warm-up duration: 0 * hours 0 * minutes 0 * seconds   Run duration: 0 * hours 10 * minutes 0 * seconds   Sampling rate: 5 * seconds	ľ		
Network Mix Counter Sets Run Settings	Description:			
	Maximum error details: 100 😴 Validation level: Low - invoke validation rules marked low			
	< Previous Next > Finish Cancel			

## **Creating load test – Visual Studio**



## **Executing load test – Visual Studio**



### Glossary

- Transactions A transaction is defined as a Web page request that results in a Web browser output display
- Response Time Response time is the accumulation of time associated with processing requests (service time) and waiting for service (queue time).
- Queue Time Queue time is the amount of time spent waiting to receive service
- Service Time Service time is the time required by a device (or server) to service a request
- Throughput Throughput is the rate at which some amount of work is being performed
- Utilization Utilization measures the fraction of time a device is busy servicing a request and is usually reported as a percentage

### Glossary

- Scalability Scalability is the ability of a computer system to adapt to an increasing load demand while providing minimal performance degradation
- Performance Performance is related to response time and is based on a user's observation of how well a system is performing a given task.
- Performance Testing Used to determine how fast some aspect of a system performs under a particular workload
- Stress Testing Stress testing is a form of testing that is used to determine the stability of a given system or entity
- Load Testing Load testing is the process of putting demand on a system or device and measuring its response