



Visual Studio
ALM Days 2012



Load Testing mit Visual Studio 2012

Frank Maar, Microsoft Deutschland GmbH

Übersicht

Gründe und Zeitpunkt

Einzeltests erstellen

Einzeltests in Lasttests verwenden

Windows Azure zur Lasterzeugung

Gründe

- (Fast) keine Anwendung skaliert linear
- Hohe Benutzeranzahl schlechtere Antwortzeiten



Architekturanforderungen prüfen

PERFORMANCE

Maximum
3 seconds
average page
response
time

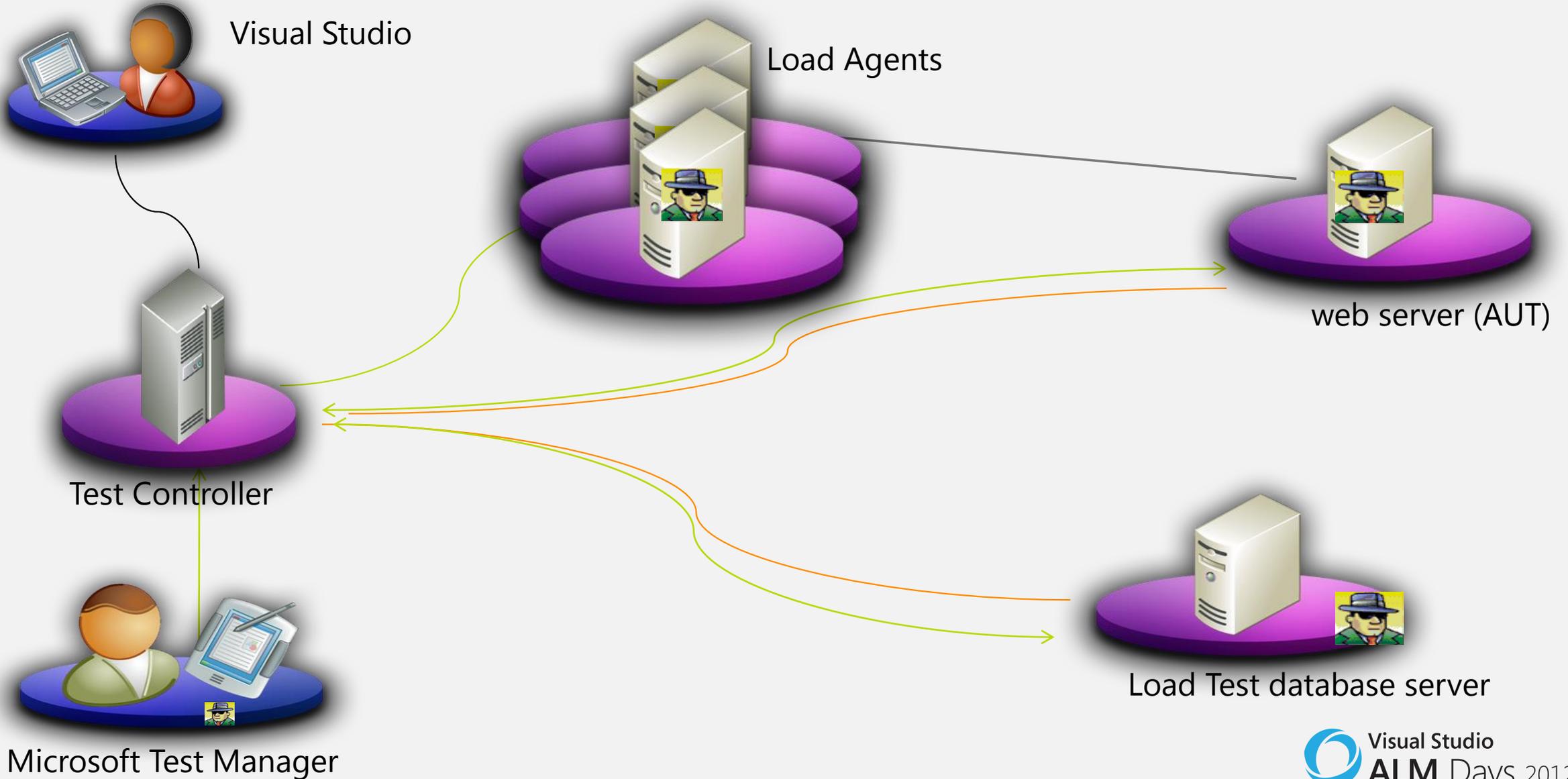
Minimum
25 orders per
seconds

Minimum
1200
concurrent
users with no
timeouts

1 new front
end server
adds 20%
more users

- Zeitpunkt:
- Möglichst früh und regelmäßig
- Einfach beginnen und mitwachsen lassen

Visual Studio 2012 Lasttest Architektur



Einzeltests

WebTests

CodedUITests

Unittests

Datenbanktests

WebTest-Erstellung

The screenshot displays the Web Test Recorder interface. On the left, a tree view shows recorded requests to `http://www.fabrikam.com/ServiceTickets/GetServiceTicket` with various query string parameters like `ticketId=1` through `ticketId=5`. The main area shows a simulated web page with a dark header 'Fabrikam Fiber Support' and a yellow banner for 'Scheduled Maintenance'. Below the banner, there are two sections: 'My Tickets' with a list of ticket IDs and descriptions, and '[Assigned] Viewing Recorded Programs' with details like 'Created On: 3/31/2012 2:21 AM' and 'Assigned To: Brian Keller'.

Web Test Recorder

Record Pause Stop

- http://www.fabrikam.com/
- http://www.fabrikam.com/ServiceTickets/GetServiceTicket
 - QueryString Parameters
 - ticketId=1
- http://www.fabrikam.com/favicon.ico
- http://www.fabrikam.com/ServiceTickets/GetServiceTicket
 - QueryString Parameters
 - ticketId=2
- http://www.fabrikam.com/ServiceTickets/GetServiceTicket
 - QueryString Parameters
 - ticketId=4
- http://www.fabrikam.com/ServiceTickets/GetServiceTicket
 - QueryString Parameters
 - ticketId=5

Fabrikam Fiber Support

Scheduled Maintenance: Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

My Tickets

- [A014101] - Modem keeps resetti... ◆
- [A014102] - Internet Upload spee... ✓
- [A014103] - FabFiber is the worst... ◆
- [A014104] - changing channel by... ◆
- [A014105] - Viewing Recorded Pr... ◆

[Assigned] Viewing Recorded Programs

Created On:
3/31/2012 2:21 AM

Assigned To:
Brian Keller

WebTest-Ausführung

The screenshot shows the Visual Studio WebTest execution results and a browser view. The top part is a table of test results, and the bottom part is a browser view of the test results page.

Request	Status	Total Time	Request Time	Request ...	Response By...
http://www.fabrikam.com/	200 OK	0.060 sec	0.023 sec	0	109,350
http://www.fabrikam.com/ServiceTickets/GetServiceTicket	200 OK	0.008 sec	0.008 sec	0	268
http://www.fabrikam.com/favicon.ico	404 Not Found	0.002 sec	0.002 sec	0	3,056
http://www.fabrikam.com/ServiceTickets/GetServiceTicket	200 OK	0.008 sec	0.008 sec	0	273
http://www.fabrikam.com/ServiceTickets/GetServiceTicket	200 OK	0.007 sec	0.007 sec	0	268
http://www.fabrikam.com/ServiceTickets/GetServiceTicket	200 OK	0.008 sec	0.008 sec	0	269

The browser view shows the 'Fabrikam Fiber Support' page. A yellow banner at the top indicates 'Scheduled Maintenance'. Below it, a red error message states: 'There was an error while retrieving data from the server'. A 'My Tickets' list is visible on the left side of the page.

Dauert ein Aufruf besonders lang?

Werden viele Daten übertragen?

Treten Fehler auf?

Think Time und Response Time Goal

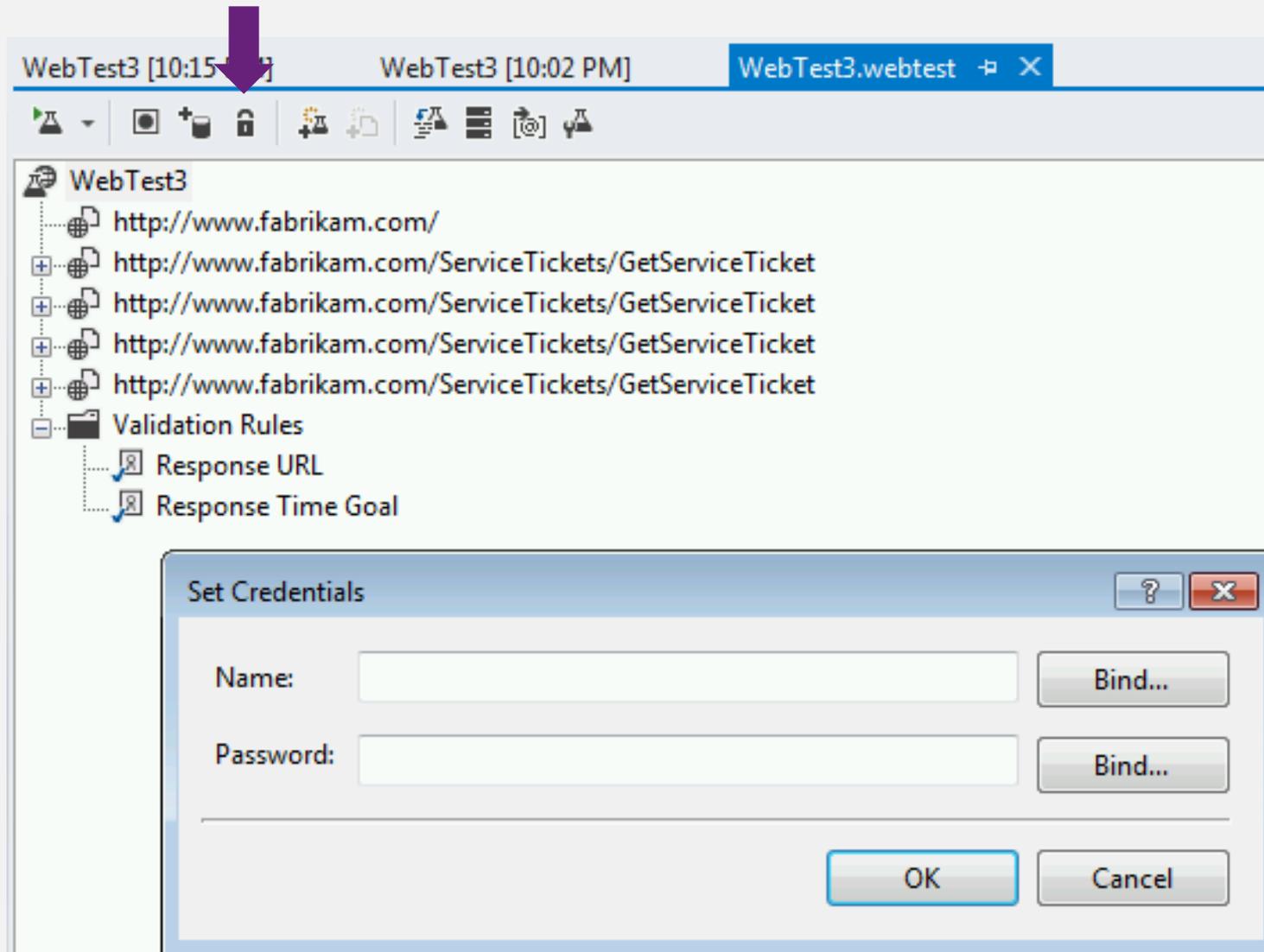
Request Details

Configure individual request properties

Request	Reporting Name	Think Time	Response Time Goal
http://www.fabrikam.com/		0	
http://www.fabrikam.com/ServiceTickets/GetServiceTicket		0	
http://www.fabrikam.com/ServiceTickets/GetServiceTicket		4	
http://www.fabrikam.com/ServiceTickets/GetServiceTicket		1	
http://www.fabrikam.com/ServiceTickets/GetServiceTicket		0	

OK Cancel

Credentials



The screenshot shows the Visual Studio interface with a web test project named 'WebTest3'. The project tree on the left includes a folder 'WebTest3' containing several test steps and a 'Validation Rules' folder. A 'Set Credentials' dialog box is open in the foreground, featuring input fields for 'Name' and 'Password', each with a 'Bind...' button, and 'OK' and 'Cancel' buttons at the bottom. A purple arrow points to the 'WebTest3' tab in the top toolbar.

WebTest3 [10:15] WebTest3 [10:02 PM] WebTest3.webtest

WebTest3

- http://www.fabrikam.com/
- http://www.fabrikam.com/ServiceTickets/GetServiceTicket
- http://www.fabrikam.com/ServiceTickets/GetServiceTicket
- http://www.fabrikam.com/ServiceTickets/GetServiceTicket
- http://www.fabrikam.com/ServiceTickets/GetServiceTicket
- Validation Rules
 - Response URL
 - Response Time Goal

Set Credentials

Name: Bind...

Password: Bind...

OK Cancel

Caching

Properties Microsoft.VisualStudio.TestTools.WebStres

Cache Control	False
Encoding	utf-8
Expected HTTP Status Code	0

Cache Control
True indicates that the requests are cached.
False indicates that the requests are not cac...

Datengetrieben

New Test Data Source Wizard

Select the type of data source

Data source name
DataSource1

Data source type

- Database
- CSV File
- XML File

Please select the type of data that will be the basis for this data source

< Previous Next > Finish Cancel

Quellcode

```
WebTest3Coded.vb*  WebTest3.webtest  WebTest3 [10:15 PM]  WebTest3 [10:02 PM]  LoadTest4.loadtest
WebTest3Coded  Run()
Public Overrides Sub Run()
    'Initialize validation rules that apply to all requests in the WebTest
    If (Me.Context.ValidationLevel >= Microsoft.VisualStudio.TestTools.WebTesting.ValidationLevel.Low) Then
        Dim validationRule1 As ValidateResponseUrl = New ValidateResponseUrl()
        AddHandler ValidateResponse, AddressOf validationRule1.Validate
    End If
    If (Me.Context.ValidationLevel >= Microsoft.VisualStudio.TestTools.WebTesting.ValidationLevel.Low) Then
        Dim validationRule2 As ValidationRuleResponseTimeGoal = New ValidationRuleResponseTimeGoal()
        validationRule2.Tolerance = 0.0R
        AddHandler ValidateResponseOnPageComplete, AddressOf validationRule2.Validate
    End If

    Dim request1 As WebTestRequest = New WebTestRequest("http://www.fabrikam.com/")
    MyBase.Send(request1)
    request1 = Nothing

    Dim request2 As WebTestRequest = New WebTestRequest("http://www.fabrikam.com/ServiceTickets/GetServiceTi
    request2.QueryStringParameters.Add("ticketId", "1", False, False)
    MyBase.Send(request2)
    request2 = Nothing

    Dim request3 As WebTestRequest = New WebTestRequest("http://www.fabrikam.com/ServiceTickets/GetServiceTi
    request3.ThinkTime = 4
    request3.QueryStringParameters.Add("ticketId", "2", False, False)
    MyBase.Send(request3)
    request3 = Nothing
```

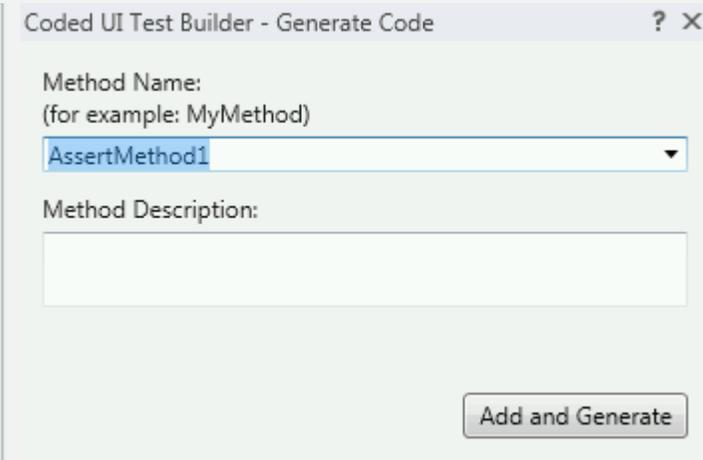
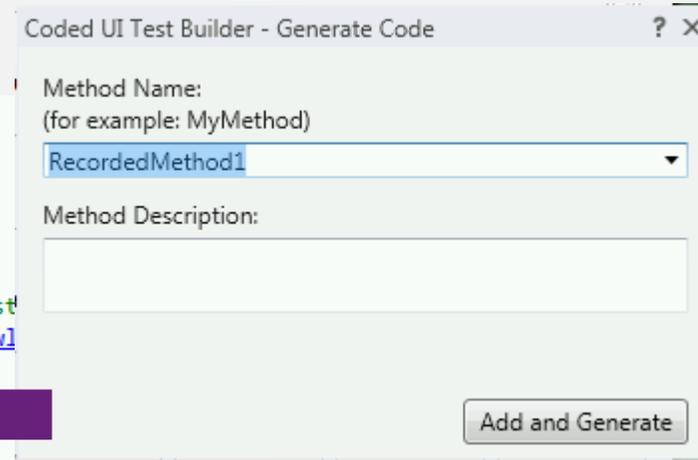
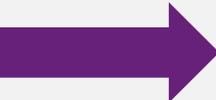
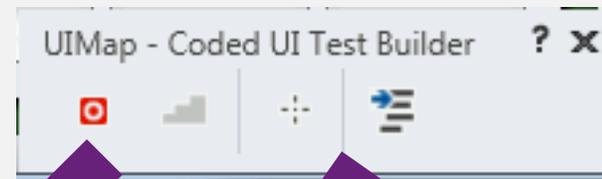
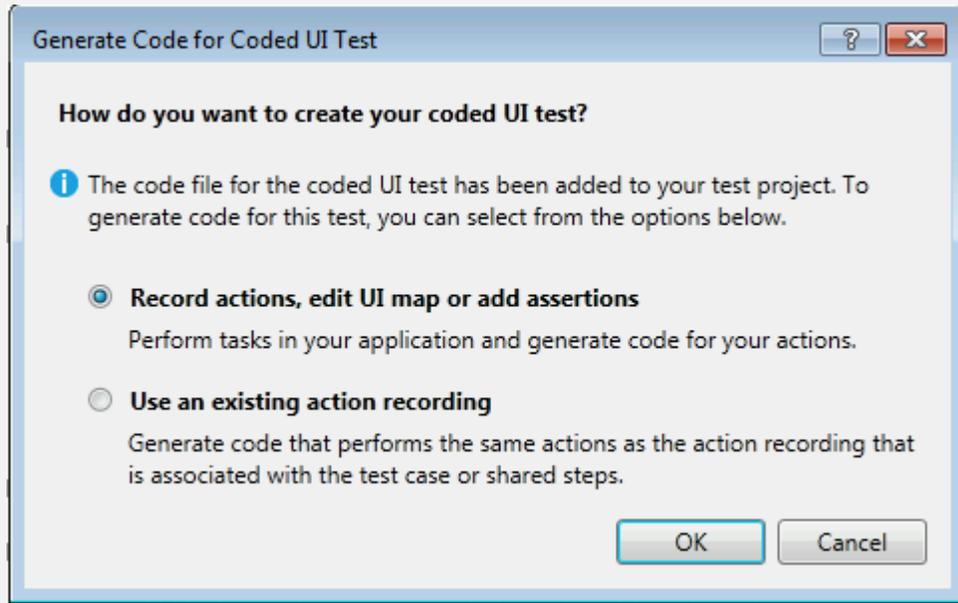
WebTest

Demo

Neu, ausführen und datengetrieben

CodedUITest

- Aufzeichnung und Abspielen von Benutzerinteraktionen



```
<CodedUITest()>
Public Class CodedUITest1
    <TestMethod()>
    Public Sub CodedUITestMethod1()
        ' To generate code for this test, select "Generate Code for Coded UI Test"
        ' For more information on generated code, see http://go.microsoft.com/fwlink
        Me.UIMap.RecordedMethod1()
        Me.UIMap.AssertMethod1()
    End Sub
End Class
```

CodedUITest

Demo

Unittests

```
[TestClass()]
public class CustomersControllerTest
{
    MockCustomerRepository mockCustomerRepo;
    CustomersController controller;

    [TestInitialize()]
    public void SetupController()
    {
        mockCustomerRepo = new MockCustomerRepository();
        controller = new CustomersController(mockCustomerRepo);
    }

    [TestMethod()]
    public void CreateInsertsCustomerAndSaves()
    {
        controller.Create(new Customer());

        Assert.IsTrue(mockCustomerRepo.IsInsertOrUpdateCalled);
        Assert.IsTrue(mockCustomerRepo.IsSaveCalled);
    }
}
```

Test Explorer

Search

Run All | Run...

Failed Tests (1)

- ✗ ScheduleActionCorre... 28 ms

Passed Tests (7)

- ✓ CreateInsertsCustom... 70 ms
- ✓ DeleteConfirmedDel... < 1 ms
- ✓ DeleteFindAndReturns... 3 ms
- ✓ EditUpdatesCustom... < 1 ms
- ✓ IndexReturnsNonNull... 3 ms
- ✓ ItShouldThrowExcept... 41 ms
- ✓ ScheduleActionRetu... 130 ms

Unittests

Demo

Datenbanktests

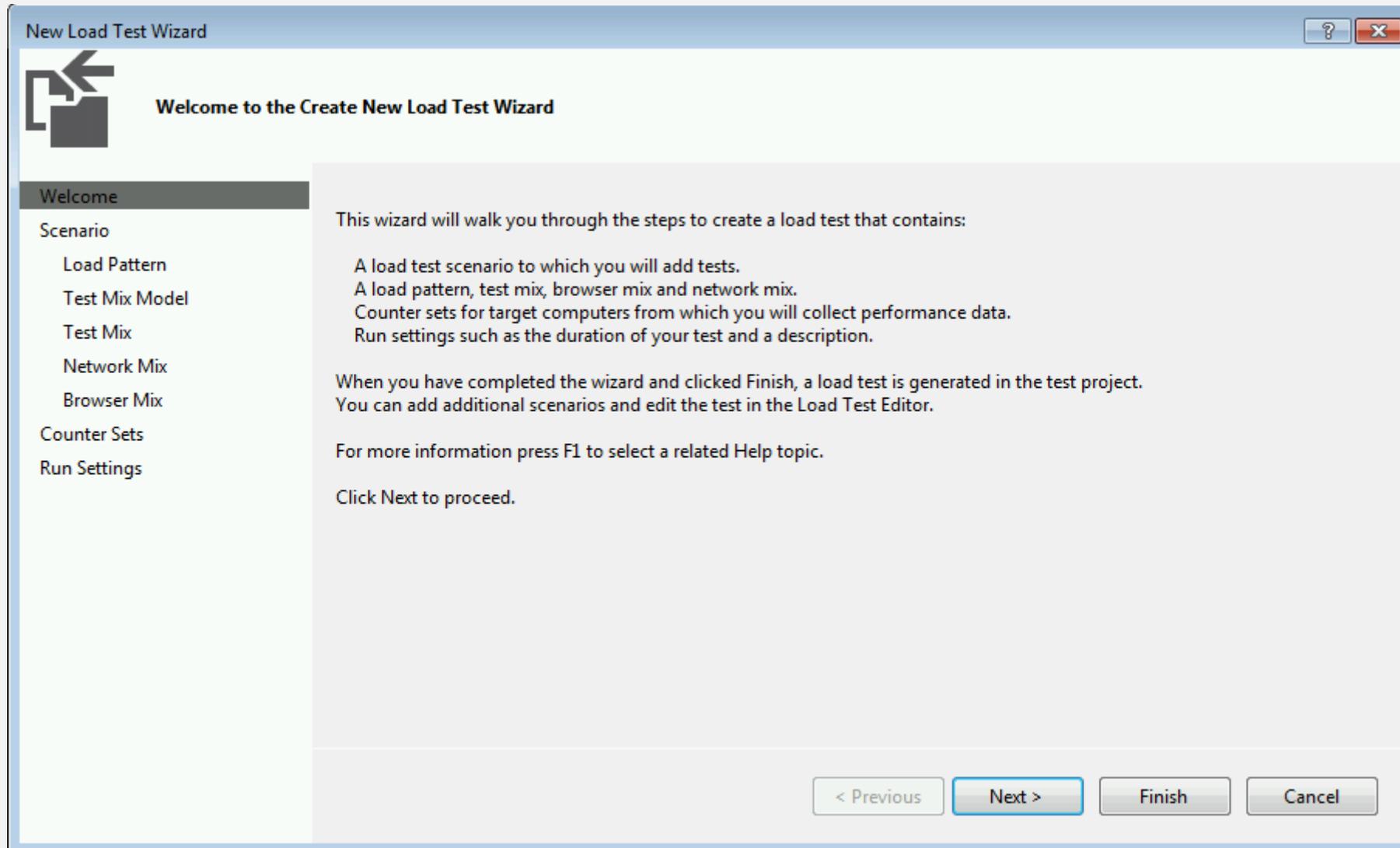
The screenshot displays the Microsoft Visual Studio IDE with the following components:

- Window Title:** FabrikamFiber.CallCenter - Microsoft Visual Studio (Administrator)
- Menu Bar:** File, Edit, View, Refactor, Project, Build, Debug, Team, SQL, Data, Tools, Unit Test, Architecture, Analyze, Window, Help
- Toolbar:** Includes icons for navigation, execution, and debugging.
- Server Explorer:** Shows a tree view with 'Home', 'Run All', and 'Failed Tests (1)'. A test 'S.' is listed with a duration of 22 ms.
- Code Editor:** Displays the source code for `UnitTest1.cs` within the `WebAndLoadTestProject1` namespace. The code includes the following elements:

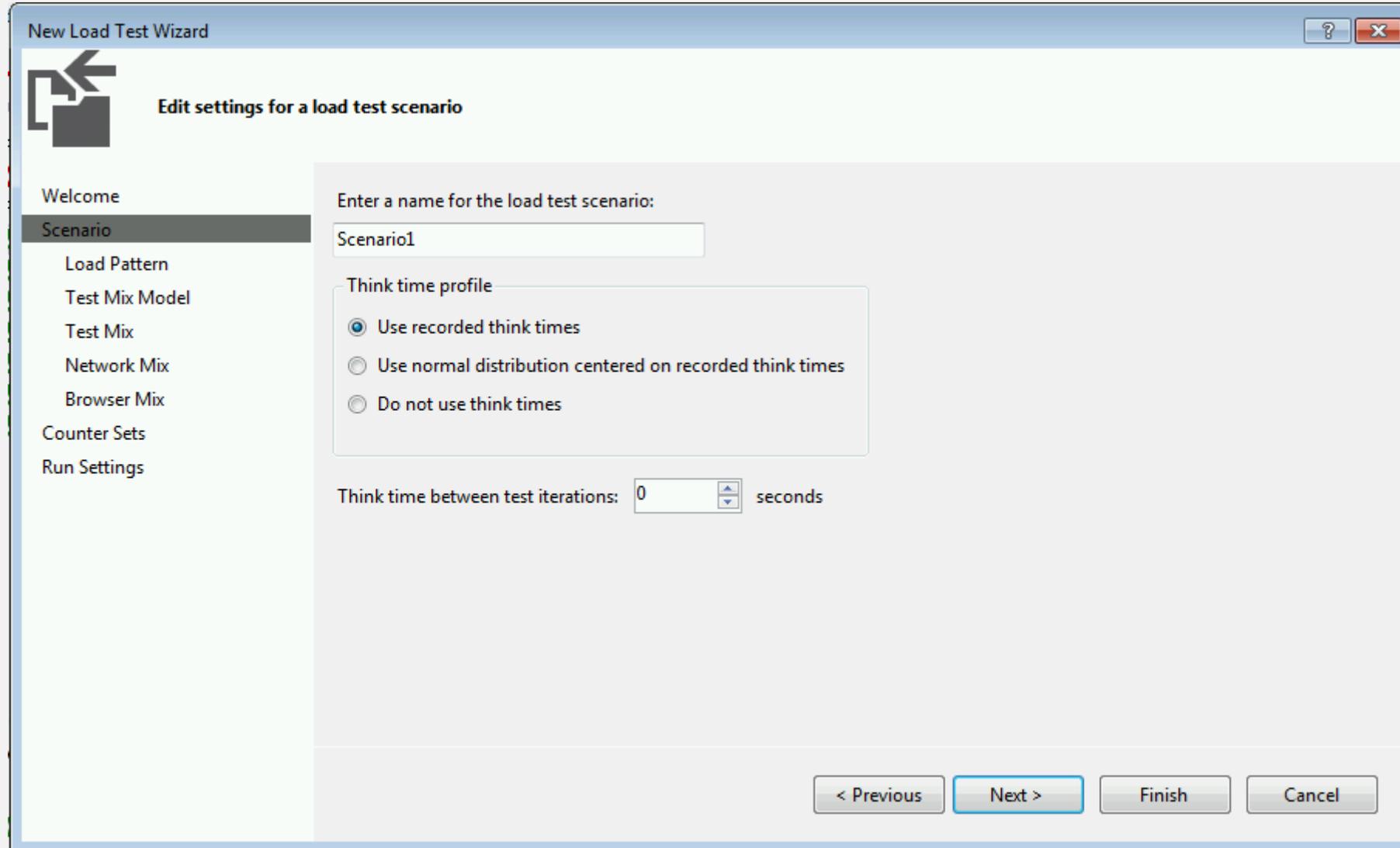
```
using System;
using Microsoft.VisualStudio.TestTools.UnitTesting;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace WebAndLoadTestProject1
{
    [TestClass]
    public class UnitTest1
    {
        [TestMethod]
        public void DatabaseSelectEmployee()
        {
            Entities db = new Entities();
            var employee = from e in db.Employees
                where e.LastName == "Keller"
                select new { e.FirstName, e.LastName };
        }
    }
}
```
- Solution Explorer:** Shows the project structure, including `LoadTest5.loadtest`, `LoadTest6.loadtest`, `Model1.edmx`, and `UnitTest1.cs`.
- Properties Window:** Currently empty.

Einzeltests in Lasttest verwenden



Think Times



Anzahl der simulierten Benutzer

New Load Test Wizard

Edit load pattern settings for a load test scenario

Welcome
Scenario
Load Pattern
Test Mix Model
Test Mix
Network Mix
Browser Mix
Counter Sets
Run Settings

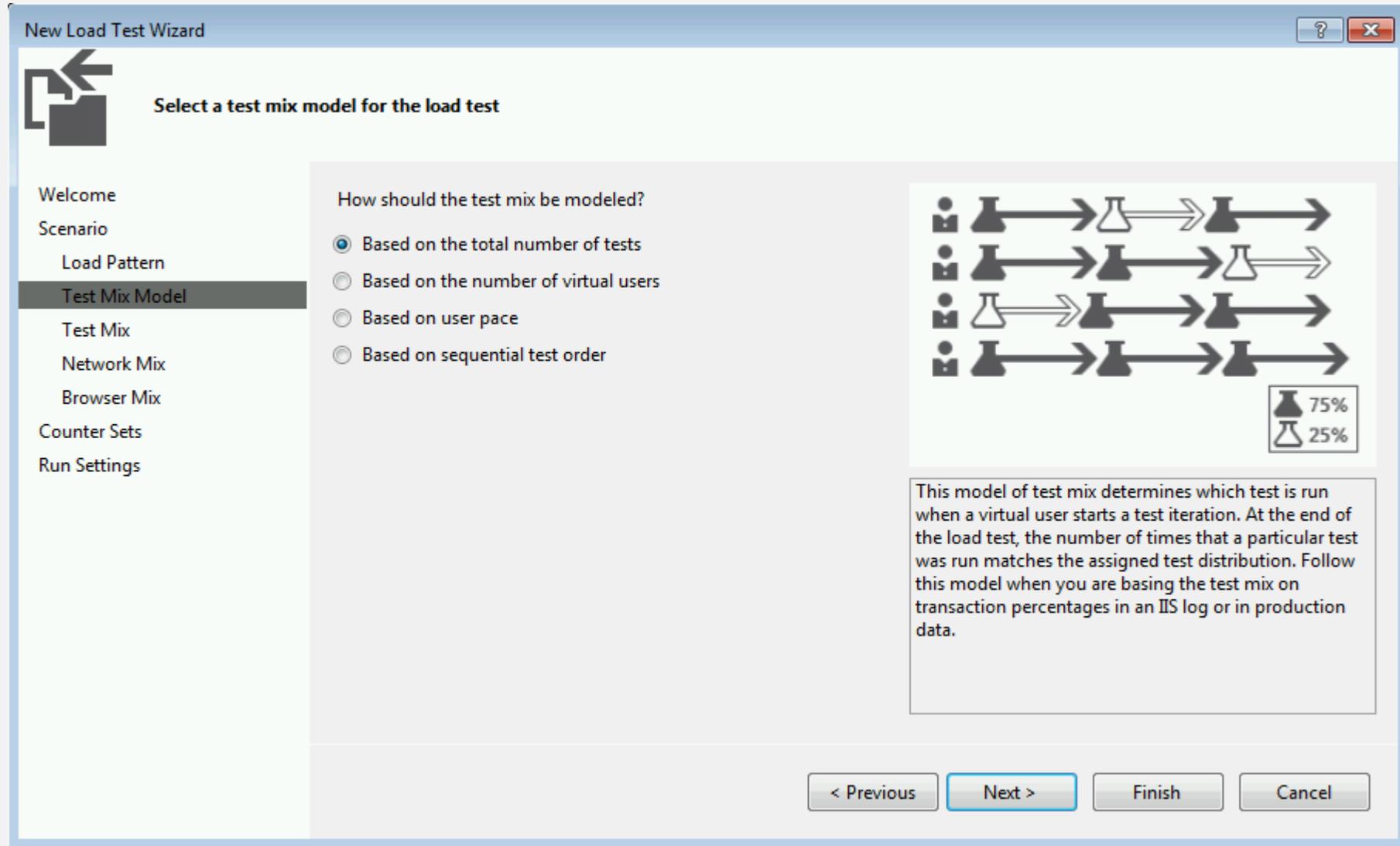
Select a load pattern for your simulated load:

Constant Load:
User Count: 25 users

Step load:
Start user count: 10 users
Step duration: 10 seconds
Step user count: 10 users/step
Maximum user count: 200 users

< Previous Next > Finish Cancel

Test mix model



Einzeltests und Verteilung

New Load Test Wizard

 Add tests to a load test scenario and edit the test mix

Welcome

Scenario

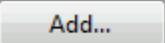
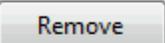
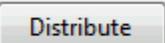
- Load Pattern
- Test Mix Model
- Test Mix**
- Network Mix
- Browser Mix

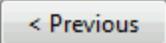
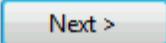
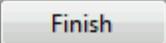
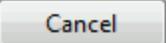
Counter Sets

Run Settings

Add one or more tests to the mix:

	Test Name	%	Distribution	
1	CreateInsertsCustomerAndSaves	32		<input checked="" type="checkbox"/>
2	DeleteConfirmedDeletesCustomerAndSaves	22		<input checked="" type="checkbox"/>
3	DeleteFindAndReturnsCustomer	6		<input type="checkbox"/>
4	EditUpdatesCustomerAndSaves	15		<input checked="" type="checkbox"/>
5	WebTest1	5		<input type="checkbox"/>
6	WebTest2	6		<input type="checkbox"/>
7	WebTest3	7		<input type="checkbox"/>
8	WebTest3Coded	7		<input type="checkbox"/>

Browser Mix

New Load Test Wizard

Add browser types to a load test scenario and edit the browser mix

Welcome

Scenario

- Load Pattern
- Test Mix Model
- Test Mix
- Network Mix
- Browser Mix**
- Counter Sets
- Run Settings

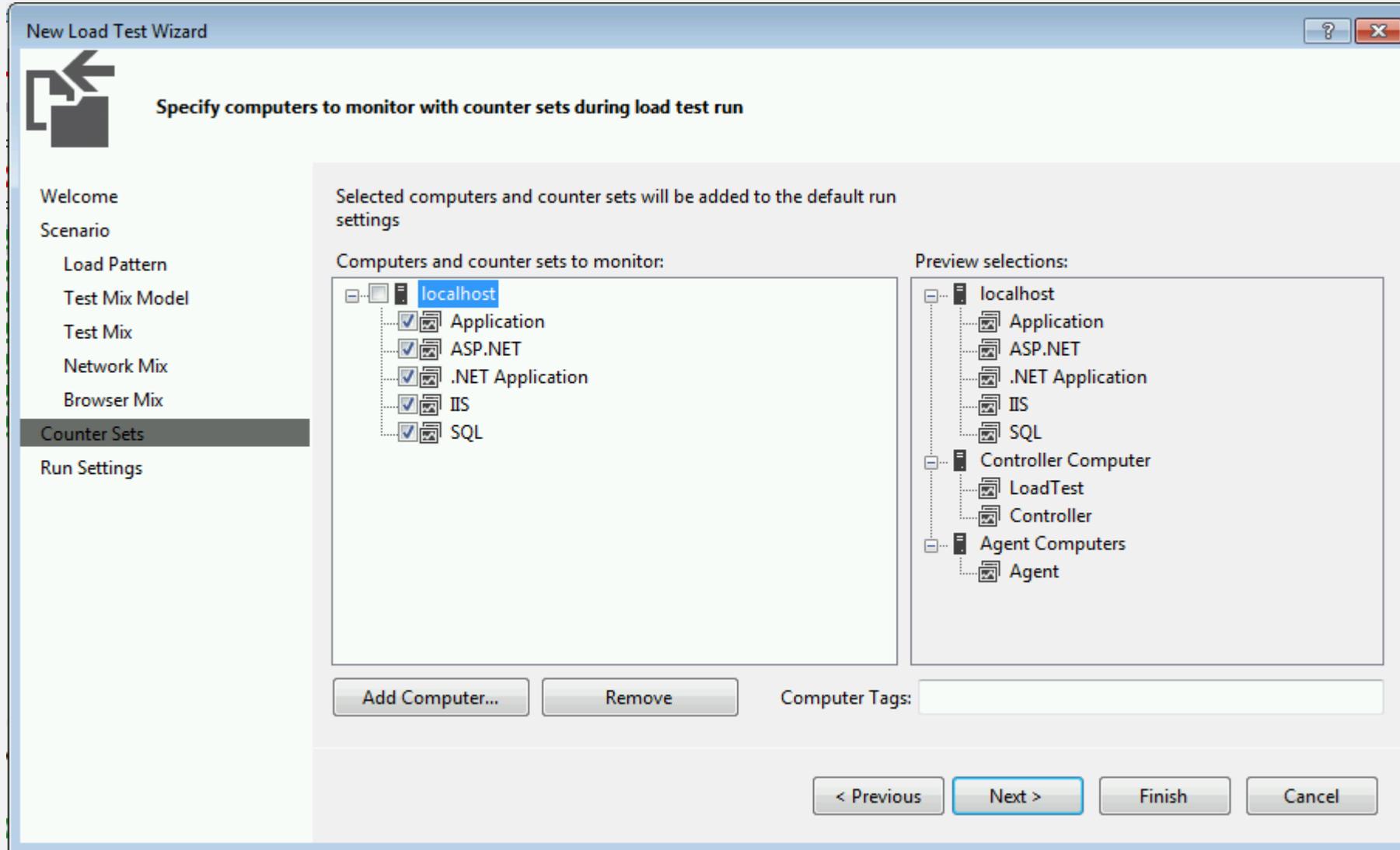
Add one or more browser types to the mix and specify a distribution:

	Browser Type	%	Distribution	
1	Internet Explorer 9.0	32	<input type="text" value="32"/>	<input checked="" type="checkbox"/>
2	Chrome 2	11	<input type="text" value="11"/>	<input checked="" type="checkbox"/>
3	Firefox 2.0	9	<input type="text" value="9"/>	<input checked="" type="checkbox"/>
4	Firefox 3.0	6	<input type="text" value="6"/>	<input checked="" type="checkbox"/>
5	Internet Explorer 10.0	42	<input type="text" value="42"/>	<input type="checkbox"/>
Total		100		

Buttons: Add, Remove, Distribute

Navigation: < Previous, Next >, Finish, Cancel

Performance Counter



Testdauer

New Load Test Wizard

Review and edit run settings for a load test

Welcome

Scenario

- Load Pattern
- Test Mix Model
- Test Mix
- Network Mix
- Browser Mix

Counter Sets

Run Settings

Specify the length of the load test by:

Load test duration

Warm-up duration (hh mm ss): 0 0 0

Run duration (hh mm ss): 0 3 0

Test iterations

Test iterations: 100

Details

Sampling rate: 5 seconds

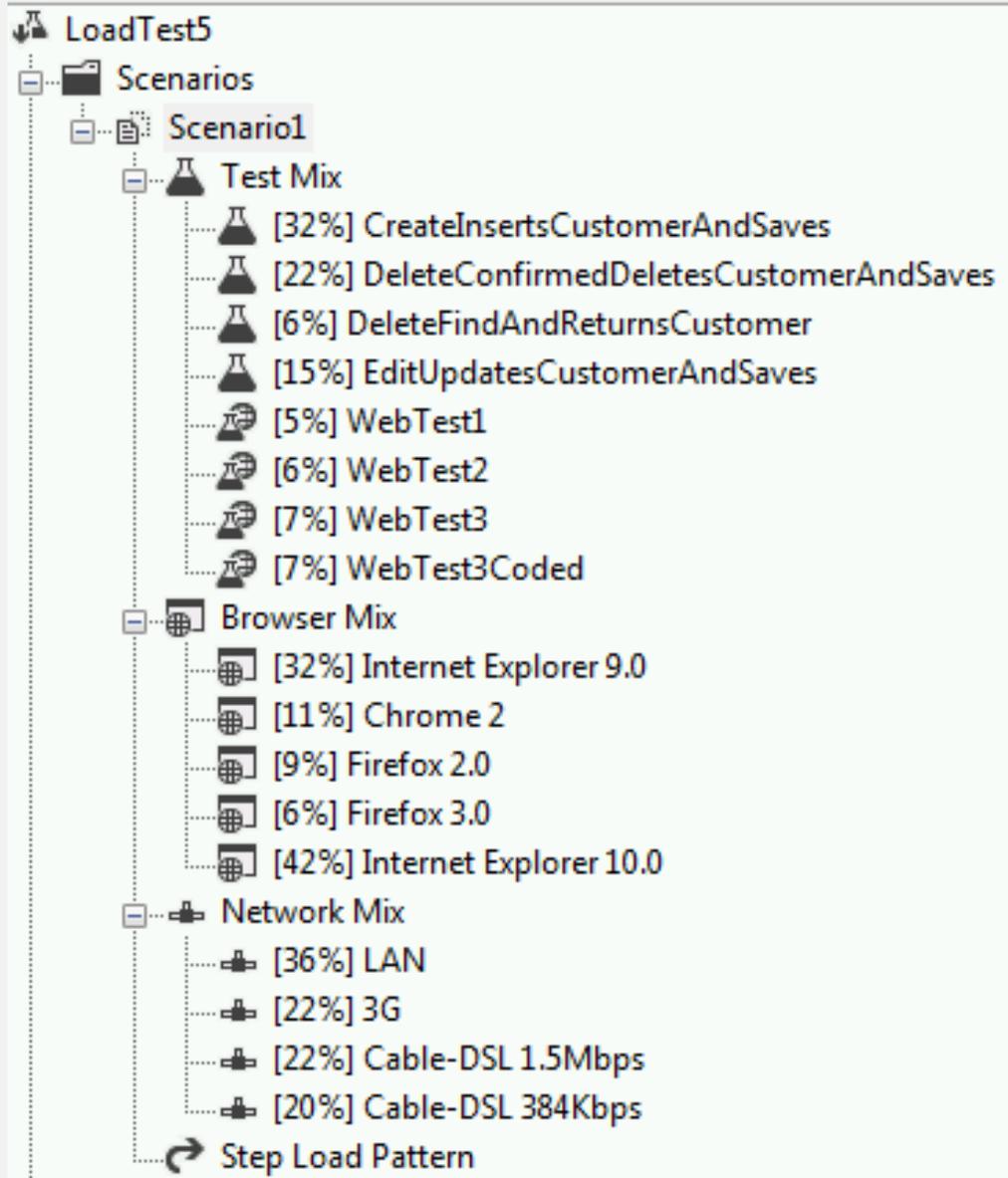
Description:

Save Log on Test Failure: True

Validation level: High - invoke all validation rules

< Previous Next > Finish Cancel

Loadtest



WebTest Connections

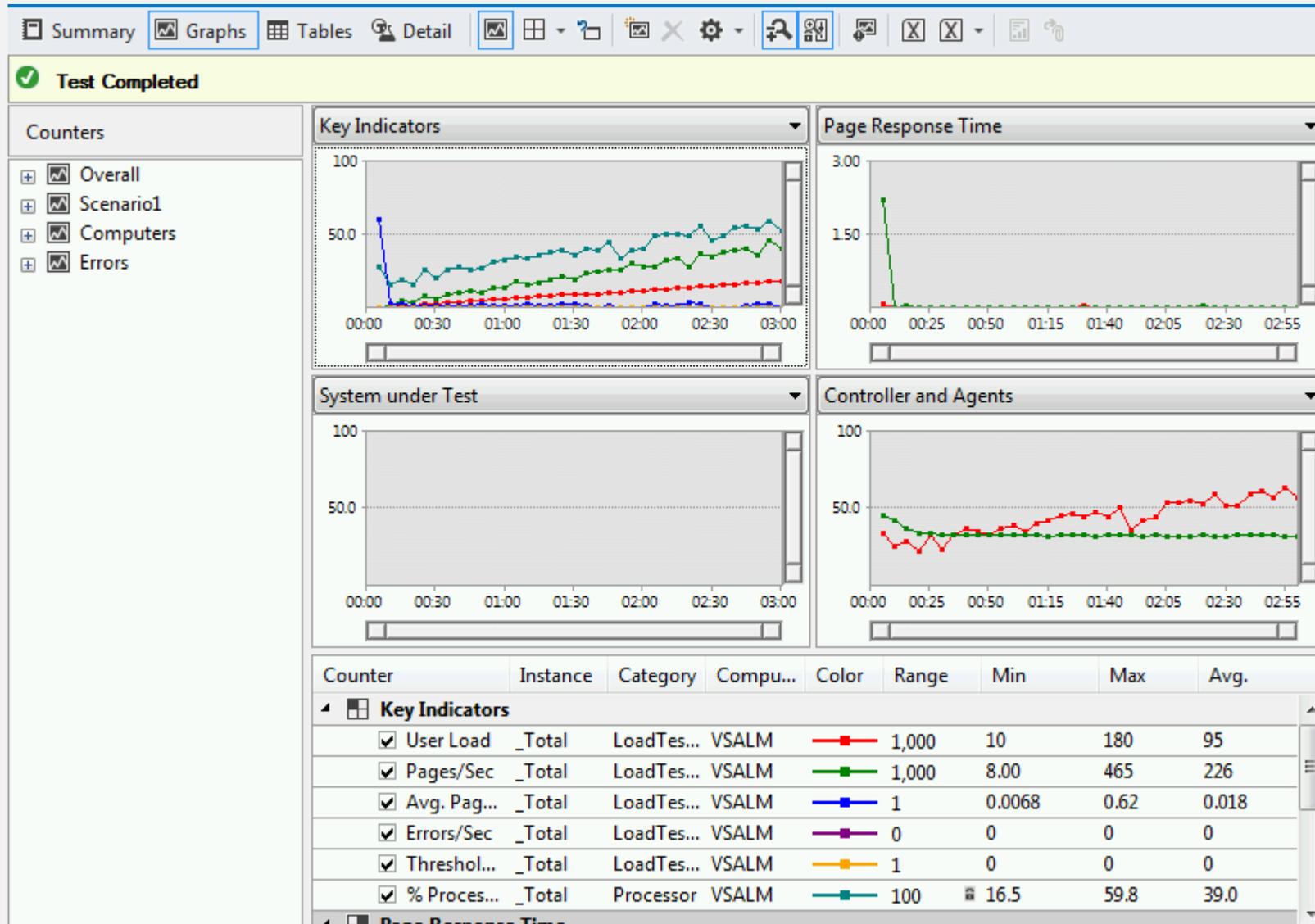
WebTest Connection Model	Connection Per User
WebTest Connection Pool Size	Connection Per User

WebTest Connection Model
Select a connection model for Web tests. 'Connection Pool' uses a pool of connections to conserve resources; 'Connection Per User': use dedicated connections for each user.

Options

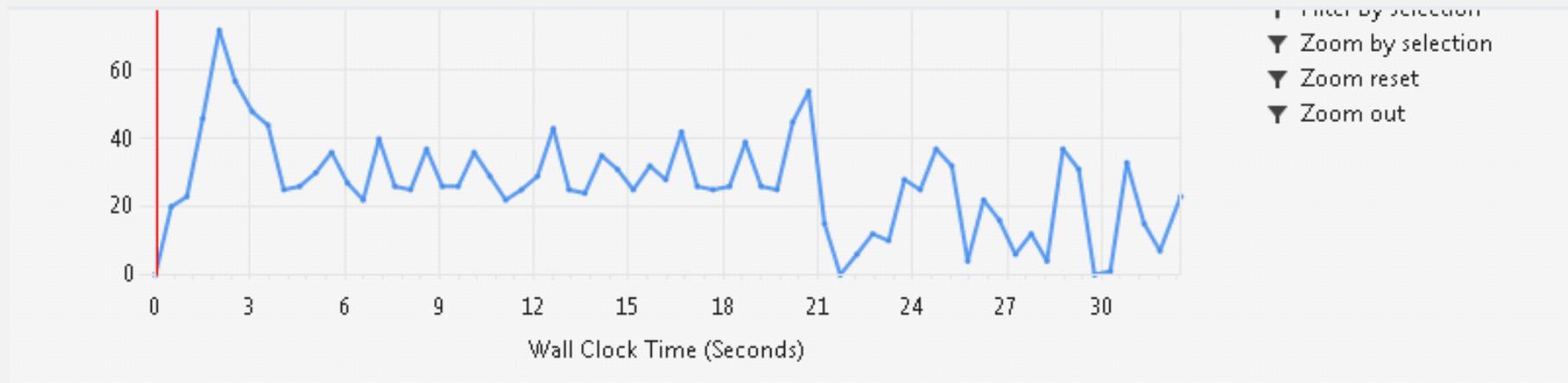
Agents to Use	
Apply Distribution to Pacing Delay	True
IP Switching	True
Maximum Test Iterations	0
Percentage of New Users	0
Think Profile	On
Think Time Between Test Iterations	0

Testergebnis



Loadtest Demo

Profiler zum Ermitteln der Engpässe



Hot Path

Function Name	Inclusive Samples %	Exclusive Samples %
WebDev.WebServer40.EXE	100.00	0.00
FabrikamFiber.Web.Controllers.HomeController.Index()	68.87	0.00
FabrikamFiber.DAL.Data.ServiceTicketRepository.AllIncluding(class System.Linq.Ex...	68.68	0.00
System.Data.Entity.DbExtensions.Include(class System.Linq.IQueryable`1...	68.62	68.62

Related Views: [Call Tree](#) [Functions](#)

Functions Doing Most Individual Work

Name	Exclusive Samples %
System.Data.Entity.DbExtensions.Include(class System.Linq.IQueryable`1<!!0>,class System.Linq.Expressions....	68.68
System.Data.Entity.DbContext..ctor(string)	4.10
System.Dynamic.UpdateDelegates.UpdateAndExecute2(class System.Runtime.CompilerServices.CallSite,!!0,!!...	3.59

Analyse bis auf Zeilenebene

Function Code View

c:\ws\brian\ff\Dev\FabrikamFiber.CallCenter\FabrikamFiber.DAL\Data\ServiceTicketRepository.cs

```
    {  
        get { return this.context.ServiceTickets; }  
    }  
  
    public IQueryable<ServiceTicket> AllIncluding(params Expression<Func<ServiceTicket, object>>[]  
    {  
        IQueryable<ServiceTicket> query = this.context.ServiceTickets;  
  
        foreach (var includeProperty in includeProperties)  
        {  
            query = query.Include(includeProperty);  
        }  
    }  
}
```

68.7 %

Profiler

Demo

Beispiel Windows Live Hotmail

- about 8 million customers in 1997 to over 260 million customers worldwide
- Web Testing and Load Testing capabilities of Visual Studio
- support 215,000 concurrent clients

Establishing a Baseline

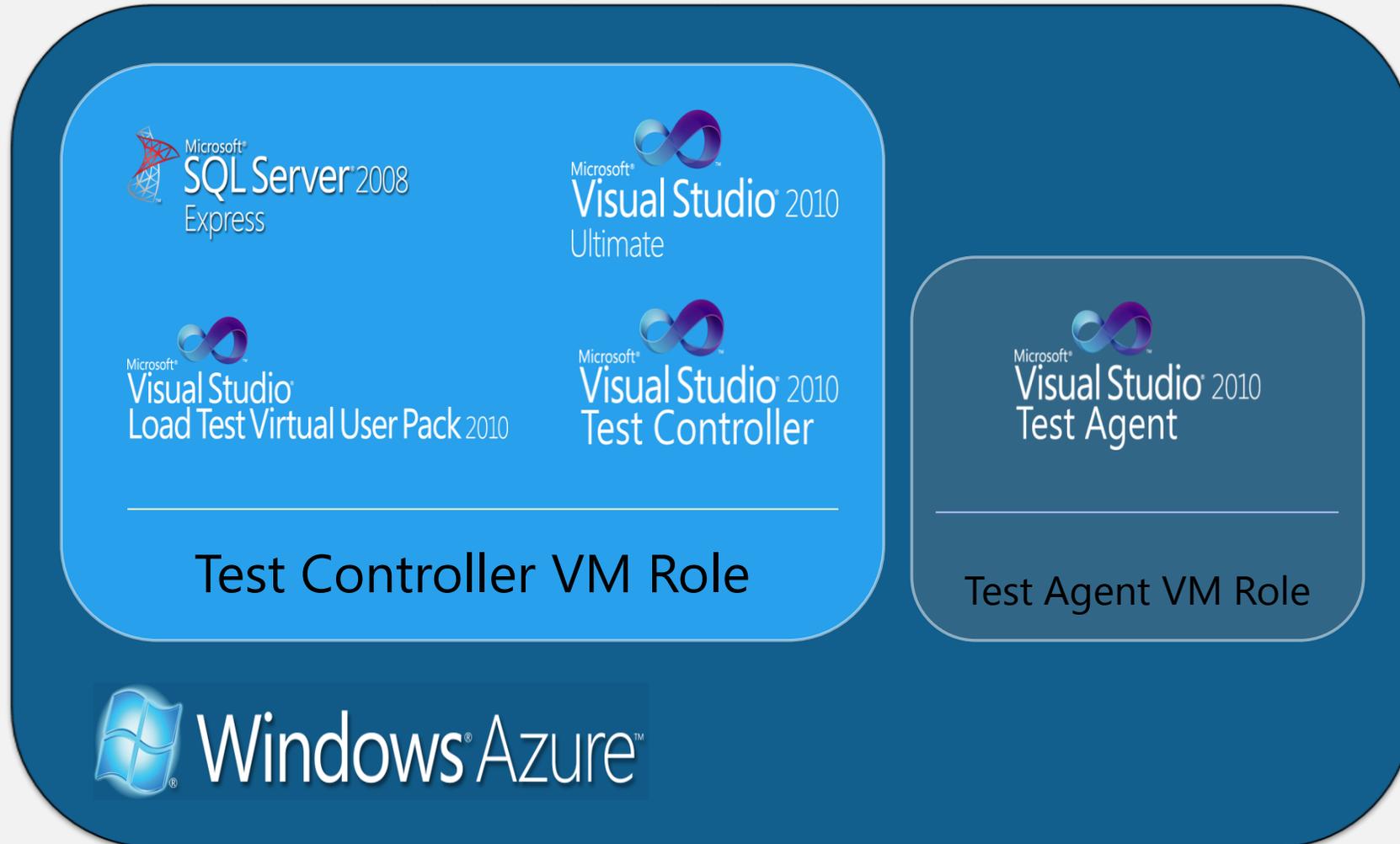
The image shows a Visual Studio environment with several overlapping windows. The top window is titled "WindowsLiveHotmail.loadtest". Below it is a "Properties" window. The main window displays the code for "WindowsLiveHotmailCoded.cs" in the "TestProject.WindowsLiveHotmailCoded" namespace, showing the "GetRequestEnumerator()" method. The code defines a class "windowsLiveHotmailCoded" that inherits from "WebTest". It includes a constructor that sets "PreAuthenticate" to true and an override of "GetRequestEnumerator" that yields two "WebTestRequest" objects. The first request is for "http://www.hot" and the second is for "http://hm.msn.". A tooltip is visible at the bottom right, explaining the "ThinkTime" property of "WebTestRequest".

```
public class windowsLiveHotmailCoded : WebTest
{
    public windowsLiveHotmailCoded()
    {
        this.PreAuthenticate = true;
    }
    public override IEnumerable<WebTestRequest> GetRequestEnumerator()
    {
        WebTestRequest request1 = new WebTestRequest("http://www.hot");
        ExtractHiddenFields rule1 = new ExtractHiddenFields();
        rule1.ContextParameterName = "1";
        request1.ExtractValues += new EventHandler<ExtractionEventArgs>();
        yield return request1;

        WebTestRequest request2 = new WebTestRequest("http://hm.msn.");
        request2.ThinkTime = 10;
        request2.Body = request2.Body;
        yield return request2;
    }
}
```

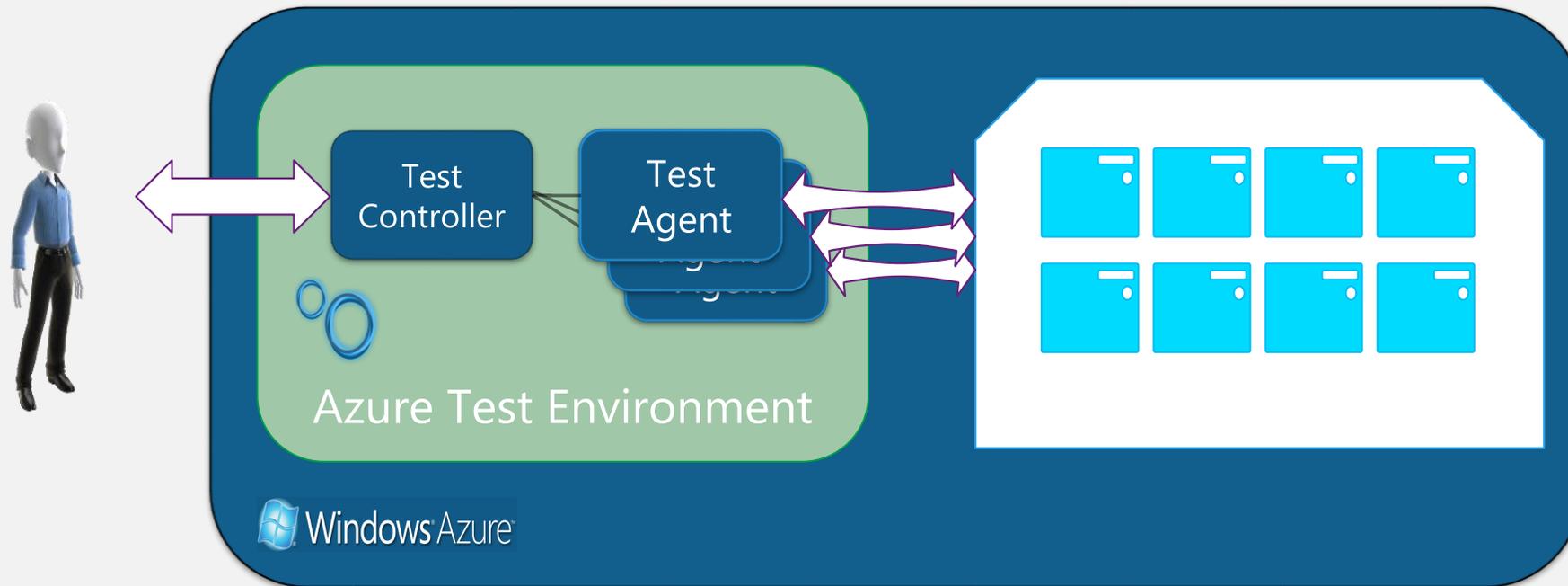
int WebTestRequest.ThinkTime
Gets or sets a value representing the think time to wait after the response is received.

ATE Ingredients



ATE = Azure Test Environment von Microsoft Consulting Services

Cloud-Anwendung testen



Weitergehende Informationen

Visual Studio Performance Testing (not so) Quick Reference Guide

A quick reference for users of the Team Testing performance features of Visual Studio

Visual Studio Performance Testing Quick Reference Guide

6/20/2011

Fragen? Danke!

Frank Maar
frank.maar@microsoft.com

Phone: +49 (30) 39097 x 335
Mobile: +49 (160) 5892428

Microsoft Deutschland GmbH
Katharina-Heinroth-Ufer 1
D-10787 Berlin

