

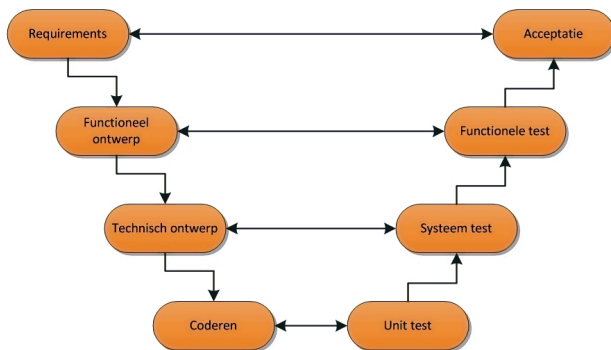
# Inrichten testomgeving met Lab Management

## DEEL 2: CONFIGURATIE EN GEBRUIK

Ewald Hofman

In .NET Magazine van december 2010 hebben we Lab Management geïntroduceerd. Dit artikel is daar een vervolg op en laat zien hoe je Lab Management configureert en gebruikt. Het eindresultaat is een platform waarin testers zelf omgevingen kunnen samenstellen en opstarten. We gaan in op het samenstellen van een regressietest en het geautomatiseerd uitvoeren daarvan.

Het testen van de applicatie doen we op de verschillende lagen in het V-model (figuur 1). De linkerkant van het V-model bevat de fases voor het implementeren van een applicatie. Elk van deze fases levert een aantal deliverables (code of document) op die je wilt valideren. Deze validaties zie je terug in de rechterkant van het V-model. Unit tests valideren de correctheid van de individuele componenten. Een unit test heeft geen afhankelijkheid van derde systemen zoals services of een database, zodat ze zijn uit te voeren zonder ons druk te maken over omgevingen.



FIGUUR 1: HET V-MODEL.

Dat wordt anders bij het uitvoeren van de systeemtest of functionele test, waarin je juist aantooit dat de onderdelen van de applicatie goed met elkaar samenwerken. Hiervoor is een omgeving nodig waarin alle componenten uit de keten geïnstalleerd zijn. Traditioneel gebruiken we een OTAP straat om in deze behoefte te voorzien. In het vorige artikel over Lab Management hebben we laten zien waarom Lab Management een beter alternatief is voor een testomgeving dan een statische OTAP.

Het doel van dit artikel is om de functionele testen voor een applicatie door de Build automatisch te laten uitvoeren in een Lab

Management testomgeving. De applicatie is een eenvoudige ASP.Net applicatie die twee getallen kan optellen. Voor het berekenen van de optelling maakt de applicatie gebruik van een WCF service.

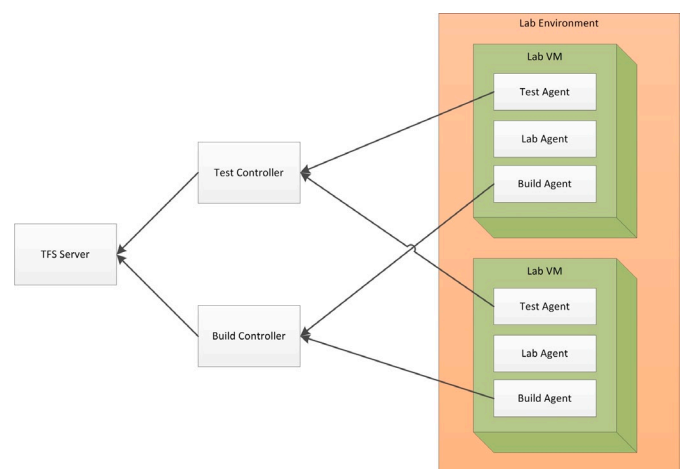
De volgende stappen moeten we uitvoeren om het doel te bereiken:

- + Installatie van de test-, build- en lab-agents
- + Opbouwen van de testomgeving
- + Configureren van de test instellingen
- + Handmatig uitvoeren van de functionele test
- + De functionele test automatiseren
- + De geautomatiseerde test opnemen in team build

### Installatie agents

Visual Studio 2010 kent drie type agents:

- + Lab-agent
- + Build-agent
- + Test-agent

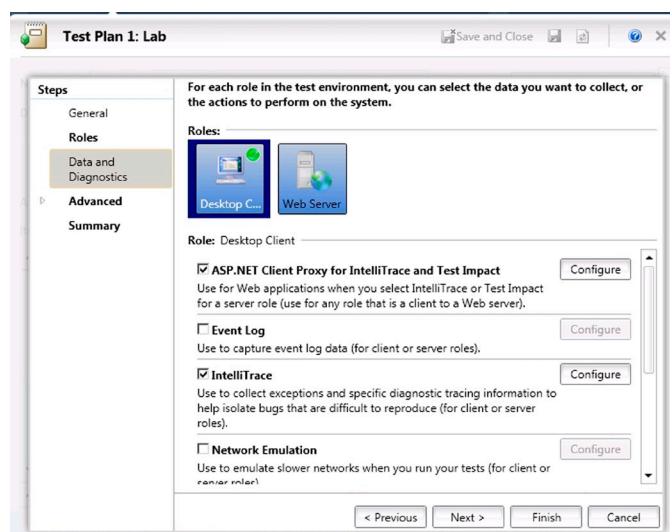


FIGUUR 2: TOPOLOGIE AGENTS IN LAB MANAGEMENT.

Een lab-agent maakt het mogelijk dat Lab Management communiceert met een machine. Verder in het artikel ga je een build aanmaken die de applicatie compileert, deployt naar de testomgeving en de test cases op de testomgeving uitvoert. De compilatie en deployment komen voor rekening van een build-agent en maakt gebruik van Windows Workflow Foundation. Een test-agent voert de test cases uit en verzamelt diagnostische informatie op de machines in de testomgeving. In figuur 2 zie je hoe de topologie van de verschillende componenten er uit ziet. In figuur 2 zie je ook een test- en een build-controller staan. Deze controllers dirigeren de communicatie van de agents richting the TFS server en bevinden zich buiten de Lab Management infrastructuur.

## Opbouwen omgeving

Lab Management maakt gebruik van System Center Virtual Machine Manager (SCVMM) voor het beheer van de omgevingen. Een omgeving is een verzameling van één of meerdere virtual machines. Je kunt van een virtual machine in SCVMM een template maken, waarbij SCVMM de sysprep functie van Windows toepast op de virtual machine. Het opbouwen van een omgeving in Lab Management doe je op basis van deze templates. In het vorige nummer hebben we gezien hoe je deze templates maakt en importeert in Lab Management. De installatie van de lab-, test- en build-agent in deze templates is aan te raden, zodat elke instantie van de template de workflow en testing capability ondersteunt. Voor de installatie van de agents kun je de VM Prep Tool gebruiken die Microsoft heeft uitgebracht die de installatie en configuratie van alle drie de agents automatisch uitvoert.



FIGUUR 3: DE TESTOMGEVING IN LAB MANAGEMENT.

Je maakt een nieuwe omgeving aan in de Environments van de Test Manager. Als je de actie New virtual environment selecteert, activeer je een wizard die je begeleidt bij het aanmaken van de omgeving. Geef de nieuwe omgeving een naam en omschrijving en in de volgende stap voeg je aan de lab-omgeving twee instanties van de Windows Server 2008 R2 template toe voor het hosten van de applicatie. De eerste machine wordt gebruikt voor de WCF services die de opteloperaties uitvoert en de tweede voor de web applicatie. Daarnaast voeg je een instantie van de Windows 7 template toe, zodat je ook een omgeving hebt waar je de browser kunt opstarten om de testen uit te voeren. In de Capabilities tab geef je aan dat alle machines in de omgeving de Test en Workflow capability willen gebruiken, zodat de Windows 7 client de test

kan uitvoeren en de twee Windows Server 2008 R2 machines een deployment kunnen uitvoeren en de diagnostische informatie kunnen verzamelen. Na het afronden van de wizard heeft Lab Management de definitie van je omgeving opgeslagen in Lab Management. Je hebt nu de definitie van de omgeving aangemaakt. Als je de testomgeving wilt gebruiken, deploy dan de omgeving.

## De action recording slaat alle interacties van de gebruiker met de gebruikersinterface op.

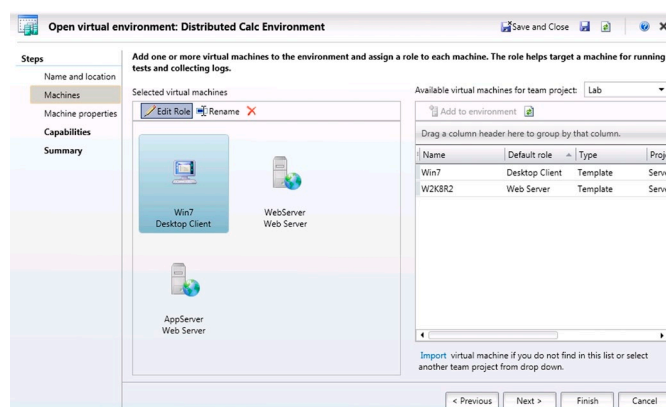
Een deploy geeft SCVMM de opdracht van alle templates uit de omgeving een virtual machine te maken. Zodra SCVMM gereed is met het initialiseren van de virtuele machines, start je de omgeving en is de testomgeving klaar voor gebruik.

## Testinstellingen

De laatste stap van het opzetten van de omgeving is het configureren van de testinstellingen van het testplan. In de test-settings definieer je welke diagnostische informatie de machines in de testomgeving moeten verzamelen tijdens het uitvoeren van de test.

De informatie die de Microsoft Test Manager voor je kan verzamelen is:

- + Action Recording
- + Video Recording
- + Test Impact
- + IntelliTrace
- + Eventlog



FIGUUR 4: DE DATA COLLECTORS VAN DE GEAUTOMATISEERDE TEST SETTINGS.

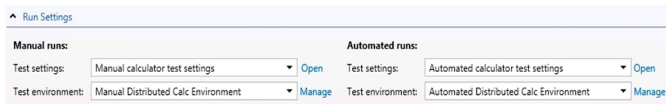
BIJ DE TEST ENVIRONMENT SELECTEER JE DE LAB OMGEVING, ZODAT DE TEST MANAGER BIJ HET UITVOEREN VAN EEN AUTOMATISCHE TEST GEBRUIKT MAAKT VAN DE OMGEVING DIE JE IN DE VORIGE STAP HEBT AANGEMAAKT.

De action recording slaat alle interacties van de gebruiker met de gebruikersinterface op. Als je eenmaal een action recording hebt, dan kun je de volgende keer dat je de test uitvoert de test (deels) automatisch af laten spelen door gebruik te maken van de action recording. Een ontwikkelaar gebruikt deze recording om de bug te reproduceren op zijn eigen omgeving en de tester gebruikt de recording voor de validatie van de bugfix.

De video recording is een screen recording van de desktop van de tester. Bij het reproduceren van de bug ziet de ontwikkelaar welke interactie de tester met de applicatie heeft gehad. Een voorbeeld is bijvoorbeeld als een tester een melding heeft gemaakt dat het sluiten van een dialoog niet goed gaat. Dit kan met het toetsenbord (escape of ALT+F4), met de OK of Annuleer knop, of met het kruisje rechtsboven in het scherm. Een applicatie kan anders reageren op elk van deze acties. Met de recording zie je welk van de mogelijkheden de tester heeft gebruikt.

Een tweetal van de nieuwe features van Visual Studio 2010 zijn de IntelliTrace en de Test Impact. De test-impact houdt tijdens het uitvoeren van een test bij welke methodes zijn aangeroepen. Als een ontwikkelaar een methode wijzigt, dan weet Visual Studio op welke testen dit impact heeft. Als je een nieuwe versie op de testomgeving installeert, dan laat de Microsoft Test Manager zien welke testen gewijzigde code hebben.

De IntelliTrace legt elke aanroep van een methode (of zelfs elke regel code als je daar voor kiest) vast. Een ontwikkelaar kan een IntelliTrace log inladen in Visual Studio en de applicatie debuggen alsof hij zelf de applicatie heeft gestart. Je kunt naar elk gewenst punt in de sessie springen en van daaruit zowel voor- als achteruit stappen in je code. In het vorige .NET magazine is Tom de Koning uitgebreid ingegaan op de mogelijkheden van IntelliTrace.



FIGUUR 5: DE TESTSETTINGS VOOR DE HANDMATIGE EN GEAUTOMISEERDE TEST.

## Uitvoeren test

Nu alles is geconfigureerd, voer je de eerste test uit. Start de test case op die twee getallen bij elkaar optelt. Voer de stappen in de test case uit: je start de browser op en navigeert naar de website op de webserver in onze Lab Management omgeving. Vul de getallen 1 en 1 in, klik op de '=' knop en valideer de output, die inderdaad 2 is. Markeer de test case als geslaagd.

De tweede test telt twee letters bij elkaar op. Bij het uitvoeren van de test zie je een `InvalidFormatException` op de pagina verschijnen, terwijl je een nette foutmelding had verwacht. Markeer de stap als gefaald en maak een nieuwe Bug work item aan. De Microsoft Test Manager vraagt alle diagnostische informatie op bij de test-agents in de testomgeving en voegt deze informatie toe aan de Bug.

Met de eerdere instellingen bevat de bug de volgende informatie:

- + De systeeminformatie van alle drie de omgevingen;
- + Een video recording van de desktop van de Windows 7 client in de Lab omgeving;
- + Een lijst van de stappen die de tester heeft uitgevoerd met per stap een referentie naar de locatie in de video recording waar de stap begint;
- + De IntelliTrace logs van de web- en applicatieserver;
- + De toevoegingen in het eventlog van de applicatie op de web- en applicatieserver.

De bug bevat op deze manier veel waardevolle informatie waarmee de ontwikkelaar snel inzicht kan krijgen wat de oorzaak van de bug is. De action recording van de test case stelt de ontwikke-

laar in staat dezelfde test stappen automatisch uit voeren. Met de IntelliTrace logs ziet de ontwikkelaar welke code de web- en applicatieserver hebben uitgevoerd.

De Microsoft Test Manager heeft ook de mogelijkheid een snapshot aan te maken van de omgeving. Hiermee sla je de huidige state van alle machines in de testomgeving op. De link naar de snapshot wordt ook toegevoegd aan de bug. Als de ontwikkelaar vermoedt dat de fout gerelateerd is aan de omgeving, gebruikt hij de link naar de snapshot om de omgeving op te starten in dezelfde state als waar de tester de bug heeft gevonden.

## CodedUI test automatiseren

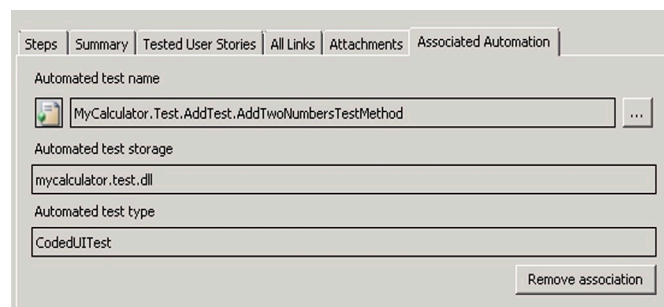
Een gouden regel in softwareontwikkeling is hoe eerder een bug wordt gevonden, hoe goedkoper het is deze op te lossen. Een deel van de bugs ontstaan op reeds werkende functionaliteit in de applicatie. Een regressietest is de set van testen waarbij je de reeds bestaande functionaliteit hertest. Deze regressietest voer je regelmatig uit, zodat tijdens de ontwikkeling van de applicatie niets 'omvalt'. Het uitvoeren van een regressietest door een tester is duur, waardoor frequentie van de regressietest niet hoog is. Als je de regressietest automatiseert, kun je de test zo vaak uitvoeren als je wilt. Het buildproces is een goede kandidaat om de regressietest aan toe te voegen, omdat het regelmatig wordt uitgevoerd en omdat de validatie van de applicatie één van de taken van het buildproces is.

Als je de testcase wilt laten uitvoeren tijdens het buildproces, moet je de testcase automatiseren. Voor het automatiseren van een testcase maak je gebruik van een nieuwe testsoort in Visual Studio genaamd CodedUI. Een CodedUI test is een unit test die met de desktop interacteert. Een ontwikkelaar maakt Coded-UI tests aan door in de User Interface interacties op te nemen, of door de action-recording te gebruiken van een bestaande testcase.

In de voorbeeldapplicatie maak je gebruik van de action-recording. Na het genereren van de CodedUI test voeg je een aantal validaties toe. Op de MSDN-website vind je meer informatie over de CodedUI tests.

Als je een regressietest toevoegt aan het buildproces, kies je een testplan uit de Microsoft Test Manager. Een testplan bestaat uit één of meer test-case work items. Je kunt een test-case automatiseren door de CodedUI test te koppelen aan de test-case in de Associated Automation tab (zie figuur 6).

Als je de testcase uitvoert via de Microsoft Test Manager of via de regressietest in het build proces, dan wordt op de achtergrond de CodedUI test uitgevoerd. De uitvoering van de CodedUI test vind plaats op de test agent in de testomgeving.



FIGUUR 6: ASSOCIATED AUTOMATION OP EEN TEST CASE.

## Team build

De volgende stap is het automatisch laten uitvoeren van de Coded-UI testen door de build. Maak eerst een build aan die de applicatie compileert en op de output op een fileshare neerzet. Daarnaast voeg je de build toe in TFS toe die de geautomatiseerde functionele testen uitvoert. Deze build gebruikt de LabTemplate build process template. Je wijzigt deze template in in de Process tab van de build definition. Open de wizard van de eigenschap Lab Process Settings. Deze wizard kent een aantal stappen, waarmee je aangeeft welke build het applicatie package aanlevert, welke testomgeving wordt gebruikt, hoe de build het package installeert en welke testen het uitvoert.

## Conclusie

Testen is een belangrijk aspect in applicatie-ontwikkeling. In de praktijk zie je dat veel organisaties worstelen met dit onderwerp: regelmatig een test herhalen is duur, meerdere testomgevingen beschikbaar stellen legt grote druk op hardware en IT Operati- ons, en de bevindingen die de testers opvoeren bevatten niet vol- doende informatie voor een ontwikkelaar.

Met het gebruik van Visual Studio 2010, Team Foundation Server, Microsoft Test Manager en Lab Management kunnen organisaties deze problemen overwinnen. In dit artikel heb je gezien hoe je Lab Management configureert zodat een tester zelf de gewenste testomgeving kan opstarten en bevindingen opvoert met veel diagnostische informatie voor een ontwikke-

laar, zoals de IntelliTrace logs en een video recording. Ook heb je een geautomatiseerd test platform geïmplementeerd waardoor de regressietest dagelijks geautomatiseerd wordt uitgevoerd.



### Links

- Lab Rangers Project: <http://go.microsoft.com/fwlink/?LinkID=206935>
- VM PrepTool: <http://code.msdn.microsoft.com/vslabmgmt>
- MSDN: <http://msdn.microsoft.com/en-us/library/dd997438.aspx>



.....  
**Ewald Hofman**, is Solution Developer bij Avanade. Hij is tevens MVP op het gebied van Visual Studio ALM. Hij blogt regelmatig over Team Foundation Server op <http://ewaldhofman.nl>.