

Softwaretesten is een complex en onoverzichtelijk gebied. Een overzicht over de eigenschappen, technieken, terminologie en tools op het gebied van testen is geen overbodige luxe. Ovum heeft geprobeerd een dergelijk overzicht te geven in het rapport *Software Testing Tools* (juli 1999). Hierin worden op basis van een raamwerk verschillende testtools geëvalueerd. In hoeverre biedt het rapport ondersteuning voor organisaties bij het selecteren van een testtool? Wat ontbreekt er aan het rapport? Hoe komen de tools uit de verf en welke tool is 'de beste'?



thema

Geen hoge score voor Ovumrapport over testen

De selectie van testing tools getest

In de verschillende definities die er van testen bestaan, komen twee aspecten steeds terug. Ten eerste wordt testen uitgevoerd om fouten in software te detecteren. Het is waarschijnlijk dat bij programmeren bugs in de software zijn geïntroduceerd. Testen kan deze bugs aantonen. Ten tweede is testen bedoeld om de werking van een softwareproduct te bepalen en te vergelijken met de verwachte werking. Testen controleert in hoeverre is voldaan aan de vastgelegde requirements maar ook aan verwachtingen die niet zijn vastgelegd in de requirements. Overigens is verwachte werking een subjectief begrip omdat verwachtingen ten aanzien van een softwareproduct kunnen verschillen. Bij testen komen vier aandachtspunten aan de orde: testentiteiten (wat wordt getest), testtechnieken (hoe wordt getest), testproces en testtools.

TESTENTITEITEN Verschillende eigenschappen of onderdelen van een softwareproduct kunnen worden getest. Opvallend is dat in de praktijk voornamelijk aandacht wordt besteed aan functionele eigenschappen: werkt het softwareproduct zoals het verwacht wordt te werken. Er kan echter ook aandacht worden besteed aan kwalitatieve eigenschappen. Een voorbeeld is het QUINT-model (Extended ISO 9126) [ZEIST]. In dit model worden kwaliteitseigenschappen van softwareproducten genoemd en gecategoriseerd onder de noemers Functionality, Reliability, Usability, Efficiency, Maintainability en Portability. Een voorbeeld van een kwaliteitseigenschap onder Maintainability is Analyseability waaronder wordt verstaan de inspanning nodig voor het diagnosticeren van de oorzaak van fouten of identificatie van delen van de soft-

ware die aangepast moeten worden. Een voorbeeld van een kwaliteitseigenschap onder Reliability is Recoverability waar wordt gekeken in hoeverre een softwareproduct in staat is zich te herstellen naar een acceptabel performanceniveau. Hierbij gaat het om herstel van zowel functionaliteit als data.

Naast de verschillende eigenschappen zijn vier onderdelen te onderscheiden waarop getest kan worden, namelijk unit, integratie, systeem en acceptatie. De termen spreken eigenlijk al voor zich. In een unittest wordt een onderdeel van een softwareproduct getest. Bij integratietesten wordt de samenwerking tussen een verzameling van units getest. Bij een systeemtest wordt het gehele product in een gesimuleerde omgeving getest. Tenslotte wordt bij de acceptatietest een product in een klantomgeving getest.

TESTTECHNIKEN De testtechnieken geven aan hóe getest kan worden. In de testwereld wordt een aantal termen vaak genoemd. De belangrijkste willen we hier kort toelichten. In de praktijk zijn veel verschillende combinaties of verfijningen van de technieken mogelijk.

- *White-box versus black-box testen* – Bij white-box testen van een softwareproduct is kennis van de code beschikbaar waarop tests gebaseerd kunnen worden. Bij black-box testen is geen kennis van de code van een softwareproduct aanwezig. Er wordt alleen gekeken hoe, bij een bepaalde invoer, het product zich gedraagt en wat de uitvoer is. Tussen beide uitersten zijn de termen grey-box of glass-box testen te noemen waarbij gebruik wordt gemaakt van globale kennis over de interne werking van een softwareproduct.

- *Regressie testen* – Met regressie testen wordt bedoeld dat wanneer een softwareproduct is gewijzigd deze opnieuw wordt getest met behulp van tests die ook zijn gebruikt vóór de wijziging. Enerzijds om te kijken of de gevonden fouten verholpen zijn en anderzijds om te controleren of de wijzigingen geen nieuwe fouten hebben veroorzaakt.
- *Statisch versus dynamisch testen* – Bij statisch testen worden tests op de code van een softwareproduct uitgevoerd. Hierbij kan code semantisch en syntactisch worden geanalyseerd. Een voorbeeld hiervan zijn compilers. Bij dynamische analyse wordt een softwareproduct getest tijdens executie.
- *Stresstesten* – Stresstesten is een vorm van testen waarbij het gedrag van een softwareproduct wordt geëvalueerd onder hoge belasting. Bijvoorbeeld wanneer het product grote hoeveelheden data krijgt te verwerken.

TESTPROCES Bij het testproces wordt onderscheid gemaakt tussen welke plaats testen heeft in het software-ontwikkelp proces en testen als organisatieproces. Bij het eerste punt kan gekeken worden naar verschillende ontwikkelmethoden. Bij bijvoorbeeld het watervalmodel gebeurt testen als laatste stap, als het product ontwikkeld is. Een ander bekend model is het V-model waarin tegenover iedere ontwikkelstap uit het watervalmodel een testfase staat. Andere ontwikkelmethoden waarin testen genoemd wordt, zijn bijvoorbeeld het spiraalmodel en DSDM.

Bij testen als organisatieproces kan gekeken worden naar de werkwijze en de omgeving waarin testen plaats vindt. Zaken die hierbij aan bod komen zijn onder andere: fasering van het testproces, selectie en gebruik middelen, kwaliteit van het testproces, meten, administratie van testcases, omgeving en terugkoppeling naar het software-ontwikkelp proces. Bekende testprocesmodellen zijn TMAP, TMM en TPI.

TESTTOOLS Het is mogelijk om bij testen gebruik te maken van tools die (delen van) de tests ondersteunen of uitvoeren. Dit is bijvoorbeeld handig bij het testen van een GUI. Een tool kan automatisch controleren of alle buttons werken. Een ander voorbeeld is een webapplicatie waarbij alle links automatisch worden gecontroleerd. Naast automatisch testen is het mogelijk tests handmatig uit te voeren. Dit is noodzakelijk wanneer tools niet voorhanden zijn of niet toegepast kunnen worden. Zo kan een tool niet weten of bij een bepaalde invoer de uitvoer correct is tenzij hem dit verteld is.

Tools kunnen in alle drie van de bovengenoemde aandachtspunten (testentiteiten, -technieken en -proces) ondersteuning bieden. Bijvoorbeeld kan in vergelijking tot testen met de hand in kortere tijd een groter deel van een product worden getest. Daarnaast kan een tool ondersteuning bieden bij het in kaart brengen en controleren op de mate van toepassing van requirements. Ook kan een tool informatie verschaffen over bijvoorbeeld perfor-

mance van een product of een omgeving simuleren waarin een product getest moet worden. Kortom, genoeg ondersteuning mogelijk die het testproces aanzienlijk kunnen versnellen en het bereik van tests kunnen vergroten.

ONVOLLEDIG Zoals in de introductie is gezegd, biedt het rapport Software Testing Tools van Ovum een raamwerk op basis waarvan softwaretesttools geëvalueerd kunnen worden en wordt een aantal tools beoordeeld. Het rapport wordt door Ovum aangeboden in de vorm van een abonnement. Het rapport heeft de volgende opbouw. Eerst geeft Ovum in het hoofdstuk Fundamentals een algemene inleiding over testen. Vervolgens wordt het raamwerk toegelicht in Guide to the evaluations. Op basis van dit raamwerk worden de tools geëvalueerd. Daarnaast is een hoofdstuk Market Analysis opgenomen waarin de huidige status en te verwachten trends worden besproken. Een aantal resterende hoofdstukken was bij de aan ons geleverde versie niet ingevuld. Het betreft hier: Comparison Tables, Briefing Papers en Glossary.

In de map die Ovum ons ter beschikking stelde, waren evaluaties aanwezig van Mercury interactive, Rational software en Compuware. Deze tools worden verderop in het artikel besproken. De complete verzameling beschikbare evaluaties bestaat volgens de Ovum website (<http://www.ovum.co.uk/>) uit: Aonix, Compuware, Cyrano, IBM, McCabe & Associates, Mercury Interactive, Rational Software, Reliable Software Technologies, RSW, Segue Software, Sun Microsystems en Technology Builders. Hierbij moet worden opgemerkt, dat de testsoftware van Sun Microsystems niet verder wordt ontwikkeld. Hierna wordt achtereenvolgens ingegaan op de hoofdstukken Fundamentals, Guide to the evaluations en de geëvalueerde tools.

FUNDAMENTALS In de inleiding van het rapport wordt een duidelijk overzicht gegeven van softwaretesten. Verschillende aspecten van testen komen aan bod waaronder testen op functionele en kwaliteitseigenschappen, verschillende technologieën (bijvoorbeeld cliënt-server, Graphical User Interface, web-based applications, testen van objectgeoriënteerde programma's), testen in relatie tot het totale software-ontwikkelp proces en de testniveau's waaronder unit, module, integratie, systeem en acceptatietest. Bij dit laatste blijft echter onduidelijk wat de niveau's inhouden en wat precies het verschil is tussen unit en module tests. Er wordt tevens nadrukkelijk gezegd dat testen een eigen proces heeft waarbij aandacht moet worden besteed aan de testing development cycle en testmanagement.

Hoewel verschillende aspecten worden behandeld, wordt geen definitie van testen gegeven. Er wordt wel gezegd dat testen het belangrijkste middel is om kwaliteit van software te garanderen. Er wordt echter ook opgemerkt dat testen demonstreert dat een softwareproduct werkt zoals het verwacht wordt te werken. Daarnaast ontbreken verwijzingen naar bestaande theorieën, modellen

en technieken zoals bijvoorbeeld de testprocesmodellen TMM en TMAP of een literatuurlijst.

GUIDE TO THE EVALUATIONS Het raamwerk op basis waarvan de tools geëvalueerd worden, is uitgebreid en lijkt volledig. Er wordt bijvoorbeeld aandacht besteed aan load and performance testing en web application testing. Daarnaast is ook veel informatie opgenomen over het product en haar leverancier. Bijvoorbeeld de versie van het geëvalueerde product, een samenvatting van de belangrijkste bevindingen, achtergrond van de leverancier, service en support die wordt geleverd en prijsverhouding. Bovendien wordt aandacht besteed aan de wijzigingen die sinds de evaluatie zijn doorgevoerd.

Er zijn echter over dit raamwerk wat opmerkingen te maken. De belangrijkste opmerking is dat de criteria waarop een tool wordt geëvalueerd erg gedetailleerd zijn en lijken te zijn afgeleid van de geëvalueerde tools. Vaak worden criteria genoemd die ingaan op de praktische aspecten van een tool. Bijvoorbeeld of de tool pause- en replay-functionaliteit biedt, grafische weergave van de resultaten ondersteunt, user-defined formats mogelijk maakt en of een glossary is opgenomen. Zo zijn er lijsten met in totaal honderden aspecten. De gedetailleerde criteria maken de evaluaties onoverzichtelijk, zodat het moeilijk is een geschikte tool te selecteren. Daarbij ontbreekt een afweging die aangeeft hoe belangrijk de criteria zijn.

Het is interessanter om te weten wat een tool kan dan in te gaan op de details van hoe de tool werkt. Het is interessant om te weten dat een tool automatisch rapporten genereert waarin testresultaten overzichtelijk zijn beschreven maar niet of de tool een grafische weergave van de resultaten ondersteunt of dat de tool user-defined formats mogelijk maakt. De meer interessante aspecten worden hierdoor onderbelicht. Is de tool bijvoorbeeld meer geschikt voor white-box of black-box testen? Is het tool geschikt voor unit of integratie testen? Kan de tool gebruikt worden voor het testen van OO-applicaties? In welk organisatieproces of situatie kan een tool het beste worden gebruikt?

Verder is de indeling die wordt gehanteerd niet altijd logisch. Zo wordt bijvoorbeeld coverage analyse genoemd onder functioneel testen. Met coverage analyse wordt bepaald welke delen van een softwareproduct getest zijn. Deze analyse is echter niet alleen van toepassing op functioneel testen. Ook unit testen wordt onterecht door de samenstellers van het rapport onder functioneel testen geschaard.

EVALUATIONS Een aantal tools uit het rapport wordt hier besproken en vergeleken. Dit zijn achtereenvolgens Compuware (QACenter, version 4), Mercury (TestSuite Enterprise, version 6) en Rational (Rational Suite TestStudio en Rational Suite PerformanceStudio, beide versie 1.1).

- *Compuware QACenter*

Compuware heeft een verzameling van testtools ont-

wikkeld die ieder een apart onderdeel van testen dekken. Zo is er een management en planningtool (QADirector) waar vanuit tests uitgevoerd en gemanaged en resultaten opgeslagen kunnen worden. Een tool waarmee regressie testen kan worden uitgevoerd door middel van capture-replay is QARun. QALoad is een tool waarmee een aantal stresstests kan worden uitgevoerd. QALoad kan helaas niet worden aangestuurd vanuit QADirector. QACenter ondersteunt mainframes, Windows en webapplicaties. Helaas is - doordat een verzameling van tools wordt geëvalueerd - niet helemaal duidelijk welke tool welke functionaliteit biedt. Bij functional testing worden bijvoorbeeld vijf tools genoemd (QADirector, QARun, File-AID/CS, NuMega DevCenter en Eco tools). Een overzicht ontbreekt waarin is aangegeven in welke situaties welk tool het beste kan worden gebruikt.

- *Mercury*

TestSuite van Mercury Interactive, marktleider in testtools, werkt zowel onder Windows 95/98/NT 4.0 als een aantal Unix-platforms, waaronder Linux, AIX en Solaris. Een breed scala aan applicaties en diverse testvormen worden ondersteund. Functioneel en load-testing wordt goed ondersteund, en er zijn voorzieningen voor het testen van GUI en web-applicaties. Bijzonder is, dat er speciale ondersteuning is voor toepassingen ontwikkeld met behulp van pakketsoftware. Er zijn extensies voor SAP R/3, Baan, Oracle Applications en PeopleSoft. Het rapport gaat helaas alleen op de SAP-extensies in.

De drie kernproducten van Mercury Interactive zijn TestDirector, WinRunner/Xrunner en LoadRunner. TestDirector is de test management tool, WinRunner en Xrunner zijn GUI Capture/Replay tools voor Windows respectievelijk Xwindows en LoadRunner is de Load testing tool. TestSuite bevat minimaal TestDirector en WinRunner, TestSuite enterprise bevat daarnaast ook LoadRunner. WinRunner en Xrunner werken met Internet Explorer 4 en 5, Netscape Navigator en Communicator 4 en hoger. De meeste functies werken ook met Internet Explorer 3.0 en andere browsers.

- *Rational*

Rational heeft de best geïntegreerde tools. Rational Suite is een verzameling van tools van requirements analyse, ontwerp (de CASE tool Rational Rose), implementatie, functioneel testen, prestatie testen, change request management en defect rapportage. Het nadeel hiervan is dat men voor gebruik van Test Studio eigenlijk niet om de Rational ontwikkelaanpak heen kan. Rational ondersteunt vrijwel alleen het Microsoft Windows-platform (NT, 95, 98).

Positief is, dat Rational aandacht besteed aan het complete proces, met of zonder tools. Het Rational Unified Process heeft voorheen SQA Proces, een gedetailleerd en praktisch tool-onafhankelijk testproces, geïntegreerd in een complete systeemontwikkelmethode. Web-based testen is zeer beperkt. Het lijkt erop dat Rational hier nog maar net mee is begonnen. Performance Studio daaren-

tegen is accuraat en schaalbaar. Het gedrag van clients wordt precies gesimuleerd. Performance Studio neemt echter aan, dat het gedrag van het onderliggende netwerk niet hoeft te worden gesimuleerd.

NIET LOGISCH Het rapport van Ovum is voor een algemene introductie op softwaretesten geschikt. Er wordt een overzicht gegeven van de verschillende aspecten en begrippen die bij testen aan bod komen. Om op basis van het rapport een tool te selecteren is echter lastiger. De criteria waarop een tool geëvalueerd wordt, zijn sterk gericht op praktische details die voor toolselectie minder interessant zijn. Daarnaast is het raamwerk op basis waarvan toolevaluatie plaatsvindt niet altijd logisch van indeling. Bovendien ontbreekt een belangrijk element, namelijk in welke situatie welk tool het beste kan worden toegepast. Bijvoorbeeld in welk stadium van het software-ontwikkelproces, bij welke ontwikkelmethode of bij welke testtechniek. Natuurlijk is wel aangegeven in welke software-omgeving een tool kan worden gebruikt.

Hoewel unit- en integratietesten wel in de inleiding worden genoemd, wordt hiermee in de evaluatie van de testtools weinig rekening gehouden. Ook objectoriëntatie wordt alleen behandeld in de inleiding, maar de geschiktheid van de tools met betrekking tot deze technologie wordt niet bekeken.

Welke tool het beste is, is op basis van dit Ovumrapport niet aan te geven, het hangt af van de prioriteiten en

voorkeuren in de gebruikersorganisatie. Verder moet worden opgemerkt dat de vorm waarin het rapport wordt aangeboden, namelijk een abonnement, in de meeste gevallen niet praktisch is. Het aanschaffen van een rapport zal over het algemeen gebeuren op het moment dat er tools moeten worden aangeschaft. Als de tools eenmaal gekocht zijn, is er weinig behoefte aan evaluaties van andere tools die via supplementen later in het rapport verschijnen.

Hugo Boer en Willem van den Ende

zijn researchers bij het Software Engineering Research Centre

Productinformatie

Ovum Evaluates: Software Testing Tools / Graham Titterington and Eric Woods

Ovum, juli 1999 (aangevuld met supplementen)

Prijs: • Decision Maker: 2075 Euro

• Decision Maker Plus: 3195 Euro

Online bestellen is mogelijk op www.ovum.co.uk/

Literatuur

[ZEIST] B. van Zeist, P. Hendriks, R. Paulussen, J. Trienekens, Kwaliteit van softwareproducten: praktijkervaringen met een kwaliteitsmodel, Kluwer, 1996