

De mix gaat anders worden

Dré de Man

Microsoft kent zogeheten Technology Adaption Programs, ook wel TAP's genoemd. Geselecteerde klanten en partners doen in een vroeg stadium ervaringen op met een product en delen die ervaringen met Microsoft. Achmea neemt deel aan het TAP-programma voor VS 2010. Teun Goosen, Community of Practice Leader Dotnet bij Achmea, legt uit wat de beweegredenen van Achmea waren en wat de eerste ervaringen zijn.

Achmea loopt met de toepassing van Microsoft-technologie voorop. Het bedrijf gebruikt onder meer Biztalk en Workflow Foundation in combinatie met oudere systemen als AS-400 in een SOA-omgeving, met Visual Studio 2008 als ontwikkelomgeving. Achmea volgt een strategie waarbij volgens een vast patroon de nieuwste technologie gevolgd en getest wordt.

Goosen: "De grote nadruk op Microsoft-technologie komt voort uit Interpolis, nu een onderdeel van Achmea. De Interpolis-strategie vanaf 2003 was enerzijds sterk gericht op het standaardiseren op het Microsoft-platform en anderzijds op het zo veel mogelijk gebruiken van standaardpakketten. De ontwikkelingen bij Microsoft gingen en gaan erg snel en daarvan wilden we maximaal profiteren. We hebben een beleid uitgestippeld waarbij we redelijk voorop zijn gaan lopen in het toepassen van die nieuwe technologieën, vanuit de gedachte dat je daardoor in kortere tijd meer functionaliteit kunt realiseren. We wilden maximaal profiteren van wat inmiddels 14.000 ontwikkelaars bij Microsoft dag in dag uit aan het doen zijn. Als je daarmee voorop loopt, zorg je ervoor dat je enerzijds optimaal gebruik maakt en dat je legacy steeds mee blijft gaan. Daarom proberen we zo veel mogelijk te uniformeren en dat beleid heeft zich doorgezet na de fusie met Achmea in die zin dat een aantal speerpuntplatforms benoemd zijn waar

Microsoft er één van is en we zijn continu bezig geweest om dat door te ontwikkelen."

Een van de redenen om aan het TAP-programma mee te doen was de behoefte om projecten zo goed mogelijk aan te sturen op basis van objectieve criteria.

Goosen: "In VS 2008, dat is de versie die we nu nog gebruiken, zitten al dingen als work item tracking, waardoor je rapportages kunt krijgen over waar je project nu echt staat, hoeveel openstaande fouten je nog hebt en dergelijke."

"Een bezoek van Ajoy Krishnamoorthy, een productmanager uit het Visual Studio Team, heeft echter de doorslag gegeven. Hij liet zien wat de eerste ideeën waren van wat toen nog Rosario was en dat gaf meteen een enorme klik. Dat waren precies de dingen die wij er nog bij wilden hebben: een nog verdergaand gebruik van de sturingsmogelijkheden voor de typen projectleden, uitbreiding van niet alleen de ondersteuning van het ontwikkelstuk, maar aan de voorkant van het ontwerpen en modelleren en requirementsmanagement en aan de achterkant van het testen en deployen. Je ziet nu dat die omgeving zich steeds meer integraal aan het uitbreiden is door alle projectfasen heen en voor alle deelnemers binnen zo'n project. Dat maakt het een stuk makkelijker om te leren



wat je in projecten niet goed hebt gedaan en om ervoor te zorgen dat je de volgende keer valkuilen voorkomt. Het maakt het vooral een stuk eenvoudiger om te zien waar je nu echt staat.”

Magisch

De voortgang van een project is vaak moeilijk te beoordelen bij gebrek aan objectieve gegevens.

Goosen: “Wanneer je een programmeur vraagt hoe lang hij over een klus doet, dan gebeurt er iets magisch: hij gaat naar het plafond kijken, denkt een tijdje na en noemt dan een aantal dagen. Als projectmanager heb je vaak weinig aan dat soort schattingen. Nu kun je inzichtelijk maken tegenover ontwikkelaars waar die schattingen tekort schieten. Je moet daar wel een interpretatie en een analyse aan verbinden. Je kunt niet zomaar zeggen dat de ene programmeur productiever is dan de andere of minder fouten maakt. Misschien heeft de één wel een veel moeilijker werkpakket gehad, slechtere specificaties of zijn de requirements veranderd. Dat is enorm waardevol voor ons, omdat we daardoor meer sturing en grip krijgen op die projecten. Wat Achmea vooral belangrijk vindt, is het invullen van de regiefunctie. Sommige onderdelen worden door derden gebouwd en daarbij is het prettig wanneer je hulpmiddelen hebt om de controle daarvan te vergemakkelijken. Ook bij locatieafhankelijk werken biedt het voordelen. Doordat je de tooling standaardiseert en integreert, wordt het ook eenvoudiger om de mensen te vinden die snappen hoe het werkt en de juiste kennis aan bord hebben om het toe te passen. Door het TAP-programma kunnen we er ook voor zorgen dat wanneer iets uitkomt we ook mensen hebben die het daadwerkelijk meteen kunnen gaan toepassen.”

Een belangrijke verandering voor de ontwikkelaar zal bestaan uit de manier van testen en de samenwerking met testers.

Goosen: “Testen en de integratie ervan is eigenlijk de hoofdreden waarom we uiteindelijk aan het Rosario-programma (later Visual Studio 2010) zijn gaan deelnemen. Je krijgt veel meer samenwerking door de communicatie tussen de ontwikkelaar en de tester. De tester ziet eerder wat er van hem verwacht wordt, kan beginnen met testen van bepaalde stukken software voordat ze helemaal af zijn en die stappen doorlopen die al wel af zijn. Op het moment dat er fouten optreden, wordt het work item teruggezet naar die ontwikkelaar. Hij krijgt een link mee – als we dat instellen – waarin wordt afgespeeld wat hij precies heeft gedaan tot dat die fout optrad. Je krijgt dus veel minder miscommunicatie in de zin van ‘ik kan het niet reproduceren’. Omdat testers en ontwikkelaars heel andere types zijn, is het verbeteren van de communicatie ook echt bevorderlijk voor de samenwerking. Door het toepassen van de nieuwe processen met quality gates heb je ook meer inzicht en worden alle stappen ook open. Is er bijvoorbeeld een functionele specificatie, zijn er unit tests gedaan, is er voldaan aan alle acceptatiecriteria vanuit je beheeromgeving, dat wordt ook echt vastgelegd en gelogd. Je kunt het later terughalen. Dat neemt weer een stuk vrijblijvendheid weg. Er zal ook een test-impactanalyse uitgevoerd worden. Dat wil zeggen dat een verandering in de code geanalyseerd wordt en dat er op basis daarvan aanbevolen wordt welke unit tests er uitgevoerd moeten worden. De cijfers van unit test die opgenomen zijn en van tests die eerder zijn gedraaid in verband met defects die zijn gerapporteerd die al dan niet open staan, alles wordt met elkaar gecombineerd. Wanneer je dan een defect sluit terwijl de betreffende unit tests dat niet ondersteunen, dan krijg daar ook een waarschuwing op.”



GOOSEN: “HET BLIJFT MENSENWERK EN DAT MOET OOK ZO BLIJVEN.”

U verwacht – meer nog dan bij Visual Studio 2008 – dat ook de samenwerking tussen de business en ontwikkelaars zal verbeteren?

Goosen: “Business-mensen gaan wanneer ze de grafieken zien, beter begrijpen wat er aan de hand is. Ze zien ook de effecten van het veranderen van requirements, met name als de functionaliteit van te bouwen software sterk moet worden uitgebreid.

Informatiemanagers blijken een grote behoefte te hebben aan dit soort gegevens. Gesprekken met hun klanten worden veel eenvoudiger, doordat de dingen geobjectiveerd worden. Het is echter nog steeds zo dat als je een analyse en een interpretatie hebt van wat je ziet. Daar zit een subjectief element in. Het blijft mensenwerk en dat moet ook zo blijven.”

Vinden ontwikkelaars dat extra inzicht altijd even prettig? Het zal ook vrijheid wegnemen, bijvoorbeeld om nieuwe technische zaken uit te testen die niet in de planning verdisconteerd waren.

Goosen: “We gebruiken het binnen Visual Studio Team System 2008 ook nog steeds niet in alle projecten op dezelfde manier. We zijn wel bezig één manier te definiëren, maar dat is een manier die vanuit Microsoft komt. Toen we hier aan zijn begonnen, kwamen er ook wel signalen terug als: ‘dit kost alleen maar tijd’. Maar je ziet wel dat op het moment dat mensen het gaan benutten het heel snel verdwijnt. Het wordt snel een tweede natuur om het wel te doen en je gaat ook de voordelen ervan inzien. Iets dergelijks speelde ook met betrekking tot kwaliteit. We doen een aantal standaard kwaliteitschecks en daar komt een rapportage uit: dit en dit zou je allemaal moeten aanpassen. Toen we dat tot standaard uitriepen was niet iedereen even blij, omdat er soms wel drieduizend meldingen gegenereerd werden. Uiteindelijk bleek het allemaal heel erg mee te vallen. Bij het wegwerken van foutmeldingen treedt namelijk ook een soort sneeuwbal effect op: los je één probleem op, dan verdwijnen er vaak meerdere foutmeldingen. Er zitten checks in rondom security, over je foutafhandeling, over de netheid waarmee je geprogrammeerd hebt. Code analysis – zo heet het – omvat bijna 250 regels, die je aan en uit kunt zetten en controleert de code op de compliance met die regels. We hebben ook met Micro-



Delta-N Quickstart

Snel uw ontwikkelproces op orde door de inzet van Visual Studio Team System


Delta-N helpt u om binnen 3 dagen een forse start te maken in het gebruik van Microsoft Visual Studio Team System. Het ontwikkelproces wordt doorgelicht, het product wordt geïnstalleerd, de gebruikers worden getraind en de organisatie is klaar om het te gaan gebruiken. Daardoor is de QuickStart het ideale middel om het ontwikkelproces te professionaliseren.

De QuickStart bestaat uit de volgende onderdelen:

- **Pre-QuickStart**
- **Intake & Analyse**
(hard-en software advies, stappenplan)
- **Installatie**
- **Implementatie en Training**

Na deze implementatie staat er een basis Visual Studio Team System met getrainde gebruikers. Niet inbegrepen zijn migraties en de "After the QuickStart" dagen.

De Delta-N QuickStart wordt aangeboden voor een fixed-price.


Microsoft
Visual Studio
Team System

Wilt u meer weten over Visual Studio Team System ga dan naar één van de volgende MSDN InTracks die in samenwerking met Microsoft worden georganiseerd op de volgende data:

11 maart
8 april
13 mei

Locatie: **Delta-N te Den Haag**
Tijdstip: **09:30 - 16:30 uur**

Voor aanvullende informatie over de QuickStart en de MSDN intracks kunt u contact opnemen met Delta-N via telefoon: 070-3575700 of via de website:

www.delta-n.nl


Delta-N
IT with brains

Delta-N
Laan van Waalhaven 287
2497 GL Den Haag

Delta-N is een van de vijf Microsoft InnerCircle partners op het gebied van Visual Studio Team System



HET ACHMEA-GEBOUW ZIET ER BIJZONDER FUTURISTISCH UIT. DE LICHTENDE RING WEKT – WAARSCHIJNLIJK ONBEDOELD – ASSOCIATIES MET DE ZANDLOPER VAN MICROSOFT VISTA.

soft gesproken over uitbreiding van die regels. Heel leuk is dat we in 2010 in staat zullen zijn die grote verzameling van regels te organiseren. Je kunt ze dan op verschillende momenten van het project inzetten, en bij develop-momenten veel minder regels toevoegen dan bijvoorbeeld op build-momenten.”

Ook op het gebied van architectuur zal Visual Studio 2010 duidelijke voordelen gaan bieden. Te denken valt daarbij aan het gebruik van UML. Een belangrijk hulpmiddel bij het vasthouden aan een bepaalde architectuur bestaat uit een systeem van regels en waarschuwingen, te vergelijken met dat voor security en softwarekwaliteit. Daarmee wordt het mogelijk om lagen te valideren op de architectuur. Wanneer een ontwikkelaar dan bijvoorbeeld besluit de API van de database rechtstreeks aan te roepen in plaats van volgens afspraak vanuit de presentatielaag via de servicelaag naar de database te gaan, genereert dat een foutmelding.

Goosen: “Je ziet dat de infrastructuur bij een bedrijf als het onze behoorlijk complex wordt en dat het voor een ontwikkelaar moeilijk te doorzien is waar iets nu precies gaat draaien en welke keuzes hij in verband daarmee moet maken. Daar helpt dit natuurlijk wel bij, omdat het meer gericht is op een standaardmanier van implementeren.”

Het programma gaf ook inzicht in een trend die goed aansluit bij de strategische beslissingen van Achmea met betrekking tot Microsoft-systeemontwikkeling: het steeds meer gebruik maken van standaardcomponenten.

Goosen: “Het werk van de ontwikkelaar verandert. Het wordt meer iemand die bouwstenen met elkaar verbindt en daarvoor programmeeractiviteiten uitvoert dan dat hij echt heel veel zelf programmeert. Dat komt bijvoorbeeld door .Net 4.0 dat weer veel uitgebreider is geworden en waarbij veel meer functionaliteit beschikbaar is, doordat het platform als geheel is doorontwikkeld. Een Sharepoint-project is wat anders dan een maatwerkapplicatie

bouwen. We zitten daar nu midden in en zien dan ook dat je op die manier heel snel heel veel kunt bereiken. Je kunt ook niet meer alles zelf ontwikkelen. Je moet voortborduren op de grote bouwstenen die er al liggen. Het werk van de ontwikkelaar verandert, het wordt meer configureren en parametriseren. Het type werk in zo'n project verandert, het krijgt niet een werkpakket waar je drie maanden aan zit te programmeren. In een paar weken heb je een website gemaakt en een paar applicatieve zaken daarachter gerealiseerd. Dat is wel heel leuk, maar voor degenen die echt gekozen hebben voor diehard programmeren is het ook even wennen. De mix gaat anders worden.”



Het Microsoft TAP-programma verschilt wezenlijk van het inzicht dat je kunt krijgen door het downloaden van vroege softwareversies via MSDN. Gerard van der Pol, Developer Technical Specialist Developer & Platform Group bij Microsoft Nederland: “Het TAP-programma behelst veel meer dan het beschikbaar stellen van software in een vroeg stadium. Aan beide kanten zitten toegewijde mensen die met een hoge frequentie communiceren. Er zit een team binnen MS met een heel opleidingsprogramma en er zijn contactpersonen bij de TAP-deelnemer. Samen bespreken ze alles, wat zijn de plannen, wat gaat er nog niet goed, van beide kanten vereist het een behoorlijke investering en we streven naar een grote toegevoegde waarde van het programma.

Hoeveel deelnemers kent Visual Studio 2010 Tap-programma?
Van der Pol: “Wereldwijd ongeveer dertig, waarvan vier in Nederland: Achmea, Avande, InfoSupport en Sogeti. We zijn in Nederland dus zeer goed vertegenwoordigd. Het is wereldwijd maar een kleine groep en ik denk dat je er trots op mag zijn wanneer je aan zo'n programma deel kunt nemen. Ajoy Krishnamoorthy, gaf na zijn bezoek aan Achmea ook als feedback: ‘deze organisatie begrijpt echt hoe softwareontwikkeling werkt.’”

Tekst en foto's: **Dré de Man**