



**Derk-Jan de Grood helpt organisaties meer grip te krijgen op hun (test)project.**

## =Spreadsheet(Business;Risiko)

Tijdens de Eurostar-conferentie ontmoette ik Felienne Hermans van de TU Delft. Zij is gepromoveerd op het testen van spreadsheets. Een opmerkelijk onderwerp omdat er traditioneel veel aandacht uitgaat naar het testen van grote en complexe administratieve systemen, embedded software of, als we het over kleinere toepassingen hebben, mobiele applicaties. Allesbehalve spreadsheets. Waarom dan toch daarop focussen?

Volgens Felienne zijn spreadsheets de meest succesvolle applicaties aller tijden. Onderzoek wijst uit dat op 90 procent van alle computers wereldwijd een spreadsheet-programma geïnstalleerd staat en dat 95 procent van alle Amerikaanse bedrijven spreadsheets gebruikt in het financiële proces. In Nederland is dat niet veel anders.

De infiltratie van spreadsheets is enorm. Ga voor jezelf maar eens na waarop de *C-level* managers in jouw organisatie hun belangrijkste beslissingen baseren. Tien tegen een dat dat gegevens zijn afkomstig uit of gegenereerd met een spreadsheet-programma. Felienne kent zelfs een organisatie die drie miljoen euro heeft verloren omdat er per ongeluk twee velden waren omgedraaid.

Omdat spreadsheetbestanden geen echte programma's zijn en er weinig programmeerervaring nodig is om ermee aan de slag te gaan, ligt de ontwikkeling ervan vaak buiten de IT-afdeling. Het zijn typisch toepassingen die klein en eenvoudig beginnen en door de jaren heen steeds verder uitdijen. Een gemiddelde spreadsheet heeft een levensduur van vijf jaar en in die tijd ongeveer dertien verschillende gebruikers. Zij brengen de ene aanpassing na de andere aan, waardoor er al snel onopgemerkte fouten of kwetsbare constructies in de spreadsheet sluipen.

Binnen de IT is het standaard om belangrijke applicaties grondig door te lichten, maar wie test de spreadsheets? Waren zij onderdeel van het IT-landschap, dan zouden ze waarschijnlijk niet aan de aandacht van professionele testers ontsnappen. Persoonlijk ken ik echter geen organisatie die ze in haar database van configuratie-items heeft opgenomen. Sterker nog: mijn ervaring is dat de IT-afdeling vaak geen weet heeft van hun bestaan.

Tijdens een van mijn opdrachten bij een zorgverlener wilde de beheerorganisatie de netwerkschijven opnieuw indelen. De

migratie was tot in de puntjes voorbereid: van alle applicaties was geïventariseerd welke mappen ze gebruikten en waren de configuratieaanpassingen meegenomen in het migratiescript en naar alle gebruikers was zorgvuldig gecommuniceerd wat de impact zou zijn. Vol vertrouwen zetten de beheerders zich aan de migratie, tot er ineens grote paniek uitbrak: de financiële afdeling lag helemaal stil. Bleken ze daar voornamelijk te werken met Excel-sheets die verborgen, hard gecodeerde verwijzingen bevatten naar sheets die door de migratie op een andere plek waren komen te staan.

Wat kunnen we hiervan leren? De eerste les is dat er verschillende manieren zijn om een spreadsheet te bouwen. De TU Delft

### Spreadsheets moeten een integraal onderdeel zijn van het IT-landschap

heeft honderden spreadsheets verzameld en geanalyseerd. Op basis van dit wetenschappelijke onderzoek hebben Felienne en haar collega's een goed beeld van de fouten die vaak worden gemaakt maar beter kunnen worden vermeden. Hard gecodeerde paden is er daar een van. Een geautomatiseerde scan geeft aan waar in een file de zwakke plekken zitten. Deze scan ziet natuurlijk geen problemen in bijvoorbeeld een verkeerd uitgevoerde berekening, maar reduceert wel de kans op fouten.

De belangrijkste les is echter dat spreadsheets een integraal onderdeel moeten zijn van het IT-landschap. We dienen ze op te nemen in het *application lifecycle management*. Omdat ze vaak cruciaal zijn voor de bedrijfsvoering, moeten we ze goed ontwikkelen, beheren en testen. Hier hebben we vanuit de IT-afdeling nog missiewerk te verrichten. Op de barricaden dus om onze collega's uit de business bewust te maken van de risico's die ze lopen. ☺