

RECURSIVITEIT ALS KWALITEITSEIS

Een aantal maal heb ik het genoeg gehad spreker te mogen zijn op het congres Trends in BPM. Een aantal van u weet dan ook dat ik al enige tijd tamboureur op de beweging dat we in de samenleving, in de politiek en in ondernemingen de stap aan het zetten zijn van *top-down* management van knechten door bazen, naar *bottom-up* eindgebruiker-gedreven ontwikkeling met *peer2peer-review* als kwaliteitssysteem. Het schouderklopje van je *peer*. Niets is zo motiverend en sturend als dat. De pakweg 1 miljoen ZZP'ers (zelfstandigen zonder personeel) die nu in Nederland opereren, zijn een reflectie van deze beweging.

Mijn vraag in deze column is: welke concrete kwaliteits-eis moet je aan huidige en toekomstige softwarepakketten stellen zodat ze robuust zijn tegen het licht van deze ontwikkeling en ook in de aankomende tijd van perpetuele bèta's een human 2.0-waarde hebben? Waar gaat het grote verschil zitten met nu? Is er wel een groot verschil? De huidige softwarepakketten staan in schril contrast met die 'ontknechting' en, in de woorden van Jaap Peters, ondersteunen vooral de intensieve menshouderij. Ze vormen nog steeds een dwingend harnas en zijn gebaseerd op 'moeten'. Meer oog voor menselijke facetten bij de software-bouwers – of eigenlijk het in het hart sluiten van de gebruikers – is een ticket naar de toekomst voor softwarepakketten en misschien wel voor IT-dienstverlening in het algemeen. Het sluit zich aan bij de ontwikkelingen in de maatschappij en verhoedt dat de softwarepakketten een soort anachronisme worden. Dus nogmaals, welke eisen zou je dan aan een softwarepakket moeten stellen? Hoe moet een pakket eruit zien dat de transitie van moeten naar ont-moeten ondersteunt?

Mijn antwoord geef ik van twee kanten, namelijk wat je niet moet doen en wat je wel moet doen. Ik denk dat ieder pakket dat mensen reduceert tot schroefjes en moertjes in een gesloten, mechanische systeembenadering kansloos is. In dit type workflow- en ERP-pakketten, zoals ze nu in elkaar zitten, wordt het werk gefragmenteerd en worden mensen uit de context van hun werk gehaald. Dat haalt de ziel uit het werk. Fragmentatie als inrichtingsprincipe is denk ik prima voor de *material* business en voor takken van sport waar sterke formele en wettelijke eisen aan worden gesteld, de rechtbank bijvoorbeeld met zijn strafprocesrecht, maar voor de rest is het geen goed idee. Waar betekenis verdwijnt, verschijnt wantrouwen en komen controllers aan de macht. Vaak is dat een goed idee, maar niet altijd.

Een kwaliteitseis die ik wel zou willen introduceren is *recursiviteit*, zeg maar het 'Drosteblikjes-effect'. Vanuit de antropologie weten we dat een sociaal systeem alleen dan levensvatbaar is, als het voldoet aan de eis dat alle subsystemen ervan in principe dezelfde functionaliteiten hebben als het geheel. Brengen we dit principe over naar de softwarewereld, dan hebben we behoefte aan systemen die ons niet alleen helpen in onze baan maar die ook de context en onze rol in de waardeketen helder maken. Zo'n systeem heeft dus niet alleen waarde voor de gehele onderneming, maar ook voor alle individuen daarbinnen. 'Upwards- en downwards-compatible' zeg maar. Collega's die met zo'n systeem werken snappen hun bijdrage aan het geheel en dat geeft betekenis.

We hebben behoefte aan systemen die ons niet langer meer vastzetten in onze functiebeschrijvingen. Dat is de 'schroefjes-en-moertjes-benadering'. We hebben behoefte aan *human centric* systemen die onze gehele competentieset ter beschikking brengen van de groep of de markt waarvoor wij kiezen. LinkedIn is bijvoorbeeld recursief. U definieert zichzelf binnen uw eigen groep. En die groep definieert zichzelf weer binnen een ander verband. Enzovoort. ERP is niet recursief. Workflow Management is in principe niet recursief. De arbeidsmarkt die zichzelf in ZZP'ers organiseert is in principe recursief, waarbij iedere collega een 'node' is in het netwerk. Een ZZP'er is een volledig bedrijf van marketing tot en met inning. Maar de groep waarvan de ZZP'er deel uitmaakt is ook een bedrijf met dezelfde functionaliteit: netwerken zijn in beginsel dus recursief. En u kent uit ervaring de betekenis van netwerken, denk ik zomaar.

In dezelfde beweging zult u gaan zien dat waar we eerst met zijn allen keken naar de *Chief Operational Officer*, er nu aandacht gaat ontstaan voor de *Chief Learning Officer* en over een paar jaar voor de *Chief Meaning Officer*.

Ik denk dat alle andere software-ontwikkelingen, hoe mooi en *cloudy* we ze ook noemen, op den duur kansloos zijn en wel omdat ze geen context en betekenis geven voor degene die er mee werkt. Softwarepakketten die recursiviteit ondersteunen leiden veel meer tot een bedrijfsvoering die niet zozeer *lean & mean* is, maar *lean & meaningful*. Denk ik.

Comments invited; reageer via mijn blog op www.businessprocess.nl

Frans van der Reep is lector eBusiness aan de Hogeschool INHolland.