

Verslag van de tweede "Conference on Software Engineering", georganiseerd door de NATO Science Committee te Rome , 27 - 31 oktober 1969.

Terwijl de eerste "Conference on Software Engineering", die vorig jaar oktober te Garmisch werd gehouden, een van de opwindendste conferenties is geweest die ik ooit heb meegemaakt, was deze een vijfdaagse marteling. Vorig jaar waren de deelnemers door de bank genomen zo hoog in hun plaatselijke hiërarchie, dat ze zich konden permitteren om eerlijk te zijn en de conferentie niet nodig hadden om erkenning te zoeken; dit keer kwam het merendeel niet om te luisteren, maar om iets te verkopen. Dat lag niet alleen aan de mensen: vorig jaar werd de omvang van de "software failure" onverbloemd aan het licht gebracht, deze conferentie was gewijd aan "wat doen we er nu aan"; terwijl vorig jaar "negatief" was geweest, moest het dit jaar "positief" zijn. Maar het is natuurlijk een beetje moeilijk om met geneesmiddelen voor een ziekte aan te komen dragen, waarvan je de symptomen gezien hebt maar waarvan je nog geen diagnose gesteld hebt. Het geheel had dan ook weg van kwakzalverij. Wat aangedragen werd of voorgesteld werd om te ontwikkelen waren allemaal van hetzelfde soort lapmiddelen, waarvan de ontoereikendheid in de jaren achter ons overtuigend is aangetoond, maar het merendeel der deelnemers was ditmaal niet in staat om dit te erkennen en bleef zorgvuldig als kat om de hete brij heenlopen.

Heel ras werd de discussie beheerst door de vruchteloze en vulgaire tegenstelling "theorie" en "praktijk". John Buxton heeft nog een dappere poging gedaan dit te doorbreken door een keer te beginnen met zoiets als: "recently I changed, as some of you may know, the ivory tower of industry for the sordid battle field of pure science...." maar het mocht niet baten.

Wat mij het meeste heeft getroffen was een sfeer van totale verlamming. Er werd cijfermateriaal aangedragen om aan te tonen dat "debugging and testing" allang meer dan vijftig procent van de kosten uitmaken en dat ze ontoereikend zijn om de fouten er uit te halen. Dus was er veel belangstelling voor "debugging aids", maar nauwelijks voor de vraag, waar de bugs vandaan kwamen.

Iedereen was het er over eens, dat bij de "Chinese Army Approach" ontstellende communicatiemoeilijkheden optreden en mensen langs elkaar gaan werken. Dus was er veel belangstelling voor automatisering van deze communicatie, maar nauwelijks voor de vraag of je het inzetten van zulke horden ook kunt omzeilen.

De verlamming werd manifest bij uitspraken "ach, eigenlijk is het goed genoeg, het hoeft niet beter: de aandelen IBM stijgen toch? En bovendien, je moet het groot zien: een software systeem is altijd ingebed in een groter geheel en daar worden ook fouten gemaakt" etc. Ik heb geregeld moeten denken aan een zin uit de kritiek uit Time op "The Organization Man" van William H. Whyte: "Is ours the society to be governed by the mediocrity of business method?".

Bij de extreme theoretici vond ik soortgelijke verlamningsverschijnselen, door formele vooroordelen: die waren als medici die weigerden om een aspirinetje tegen hoofdpijn voor te schrijven, zolang ze het elixir voor de eeuwige jeugd nog niet hadden.

Bah, bah, bah, driewerf bah!

Edsger W. Dijkstra