



## Stichting NIOC en de NIOC kennisbank

Stichting NIOC ([www.nioc.nl](http://www.nioc.nl)) stelt zich conform zijn statuten tot doel: het realiseren van congressen over informatica onderwijs en voorts al hetgeen met een en ander rechtstreeks of zijdelings verband houdt of daartoe bevorderlijk kan zijn, alles in de ruimste zin des woords.

De stichting NIOC neemt de archivering van de resultaten van de congressen voor zijn rekening. De website [www.nioc.nl](http://www.nioc.nl) ontsluit onder "Eerdere congressen" de gearchiveerde websites van eerdere congressen. De vele afzonderlijke congresbijdragen zijn opgenomen in een kennisbank die via dezelfde website onder "NIOC kennisbank" ontsloten wordt.

Op dit moment bevat de NIOC kennisbank alle bijdragen, incl. die van het laatste congres (NIOC2023, gehouden op donderdag 30 maart 2023 jl. en georganiseerd door NHL Stenden Hogeschool). Bij elkaar bijna 1500 bijdragen!

We roepen je op, na het lezen van het document dat door jou is gedownload, de auteur(s) feedback te geven. Dit kan door je te registreren als gebruiker van de NIOC kennisbank. Na registratie krijg je bericht hoe in te loggen op de NIOC kennisbank.

Het eerstvolgende NIOC vindt plaats op donderdag 27 maart 2025 in Zwolle en wordt dan georganiseerd door Hogeschool Windesheim. Kijk op [www.nioc2025.nl](http://www.nioc2025.nl) voor meer informatie.

Wil je op de hoogte blijven van de ontwikkeling rond Stichting NIOC en de NIOC kennisbank, schrijf je dan in op de nieuwsbrief via

[www.nioc.nl/nioc-kennisbank/aanmelden\\_nieuwsbrief](http://www.nioc.nl/nioc-kennisbank/aanmelden_nieuwsbrief)

Reacties over de NIOC kennisbank en de inhoud daarvan kun je richten aan de beheerder:

R. Smedinga [kennisbank@nioc.nl](mailto:kennisbank@nioc.nl).

Vermeld bij reacties jouw naam en telefoonnummer voor nader contact.



## Simulatie-onderwijs met behulp van C++ en Coroutines

Hans Kraamer

*Economische Faculteit, Erasmus Universiteit, Rotterdam*

### Inleiding

Sinds 1985 geef ik Simulatie-onderwijs aan BI-studenten. Tot 1991 betekende dit in Rotterdam het gebruik van het pakket DEMOS, geschreven in de taal SIMULA. Nu gebruik ik hetzelfde pakket, maar dan geschreven in C++.

### Simulatie

Kreutzer beschrijft drie benaderingen van Discrete Event Simulatie: event-, activity-, en process-interaction-based. De derde benadering (de PI-aanpak) heeft geleid tot het Object Oriented (OO-) paradigma. OO staat nu in de belangstelling.

Midden jaren '80 kregen vele HBO-scholen de beschikking over een SIMULA-compiler. Werkzaam op de HES-Rotterdam heb ik mij verdiept in deze taal en de door Birtwistle ontwikkelde context DEMOS. Van deze library bestaat een user-manual in boekvorm en een reference manual (1981).

### C++ en Coroutines

C++ -de opvolger van 'C with classes'- is SIMULA met een C-jasje. De semantiek van SIMULA gekoppeld aan de syntax van C. Met een gebrek. C++ kent geen RESUME - een routine die het coroutine-mechanisme ondersteunt. Nu blijkt Hansen dit op te lossen. De oplossing berust op gebruik van setjmp/longjmp en op de veronderstelling dat je een machine hebt waarop de stack geen gaten hoeft te vertonen, een eis waar een PC aan voldoet. De DEMOS-source is door mij vertaald naar C++. Er is een class-library DEMIC ontstaan.

### Nadelen bij het gebruik van C++ ten opzichte van SIMULA

Het blijkt, dat studenten -na een cursus C in een voorafgaand semester- toch gauw vergeten variabelen te initialiseren; SIMULA doet dat automatisch. Het mechanisme van parameter-overdracht in C++ van sub-class naar super-class is minder gemakkelijk op te schrijven dan in SIMULA.

De interface van de methoden WAITUNTIL, FIND en AVAIL wijkt af van die in het DEMOS-boek. Dit berust op het feit dat het CALL by NAME mechanisme van ALGOL-60 (en dus SIMULA) niet in C++ voorkomt.

### Voordelen van het gebruik van C++ boven SIMULA

- Een 'politiek' voordeel: het klinkt beter.
- Een didactisch voordeel: het geeft de student de mogelijkheid haar C-kennis van voorgaande jaren te verdiepen, en het OO-paradigma verpakt in C-kleren te ondergaan.
- Het geeft de student een Aha-Erlebnis als hij merkt dat het programmeren in C++ dezelfde antwoorden geeft als de antwoorden in het boek. Het relateert de betekenis van de taal en vertelt ook wat over de werking van een pseudo-random-generator: de antwoorden zijn precies dezelfde.
- De exe-files geproduceerd door DEMIC zijn in de orde van 90K terwijl DEMOS/PC-Simula werkt in de orde van 380K. Dit is te danken aan het feit dat DEMIC een library is waaruit de linker alleen de nuttige zaken oppikt. De library bestaat uit circa 50 modules.
- Veel studenten hebben thuis een C++-compiler en kunnen de opdrachten thuis uitwerken; dit geldt niet voor PC-Simula.

### Slot

Momenteel wordt er gewerkt aan een DEMOS C++ tekstboek.