



Stichting NIOC

Stichting NIOC en de NIOC kennisbank

Stichting NIOC (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 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 (NIOC2025, gehouden op donderdag 27 maart 2025 jl. en georganiseerd door Hogeschool Windesheim). Bij elkaar zo'n 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 18 maart 2027 in Arnhem en wordt georganiseerd door HAN University of Applied Sciences.

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

R. Smedinga kennisbank@nioc.nl.

Vermeld bij reacties jouw naam en telefoonnummer voor nader contact.

Praktisch Software Architectuur Onderwijs

Software Architectuur (SA) is een belangrijke discipline in de praktijk en het onderwijs van ICT. Door het abstracte karakter van de beslissingen en modellen binnen deze discipline blijft SA voor veel studenten ook abstract. Het is moeilijke stof om naar eigen werk te vertalen.

De abstractie van de modellen, zoals lagenmodellen en componentenmodellen, leidt er ook in de praktijk vaak toe dat de ontworpen architectuur niet goed geïmplementeerd wordt. Architecture Compliance Checking (ACC) is een benadering om de consistentie tussen de geïmplementeerde architectuur (in de code) en de bedoelde/ontworpen architectuur vast te stellen. Tools zijn hier noodzakelijk bij, omdat grote hoeveelheden code geanalyseerd moeten worden. Dit soort tools maakt veelal gebruik van statische analyse, waarmee de modulaire architectuur gevisualiseerd en gecontroleerd kan worden.

In mijn promotieonderzoek heb ik mij onder andere gericht op de ontwikkeling van een dergelijk tool, maar met als 'outstanding feature' dat veel gebruikte typen modulen (zoals subsystems, lagen en componenten) en veel gebruikte typen architectuurregels op een eenvoudige manier worden ondersteund. Hierdoor is het tool ook zeer geschikt voor het onderwijs en het is de afgelopen jaren dan ook ingezet in 2^e en 3^e jaar Informatica onderwijs aan Hogeschool Utrecht op het gebied van Software Architectuur; zowel voor architecture reconstruction als voor ACC. Studenten vinden het leuk om het tool, HUSACCT, te gebruiken en leren op een heel nieuwe manier naar geïmplementeerde architecturen te kijken (packages & classes met onderlinge dependencies). En deze implementaties te controleren op consistentie met de ontworpen architectuur. Zodoende krijgen abstracte concepten als layers, components, cohesion, coupling, encapsulation, et cetera praktische betekenis.

HUSACCT is gratis te downloaden en te gebruiken (vanaf <http://husacct.github.io/HUSACCT/>) en bijbehorend lesmateriaal wordt beschikbaar gesteld.

Een paper over HUSACCT van de wetenschappelijke conferentie 'Automated Software Engineering 2014', met daarin o.a. ook referenties van enkele andere gerelateerde papers, is te vinden op: http://koeppe.nl/publications/HUSACCT_Tool_Paper_ASE2014-Final_Author_version-2014-07-27.pdf

Achtergrond: Leo Pruijt is lecturer and researcher at the Information Systems Architecture Research Group of the HU University of Applied Sciences. Furthermore, he is PhD candidate at the University of Utrecht, also located in Utrecht. His research focuses on the design of modular software architectures, on compliance checking of modular architectures, and on tool support for software architecture compliance checking.

Doelgroep: Docenten software architectuur in het hoger onderwijs.

Motivatie: Toepassing van software architectuur in onderwijs en praktijk versterken.