

# Projectomschrijving LabView in de klas

## LabView in de klas

De computerleermiddelen in het bèta-onderwijs zijn hoognodig aan vernieuwing toe. In het veld bestaat er veel weerstand tegen de thans gebruikte software. Deze situatie is niet langer houdbaar. Ik acht de tijd rijp voor vernieuwing. Op de barricade! LabView is een goede kandidaat die de oude software kan vervangen. In een afweging met andere producten komt LabView er goed uit omdat LabView op zoveel manieren in te zetten is. LabView biedt een integrale oplossing voor de methodologie (op het gebied van ICT) van de bètavakken. [In een uitgebreide projectomschrijving zullen tzt de plussen en minnen van andere producten op een rijtje worden gezet.]

## Modern curriculum, moderne middelen

Om de onderwijsvraag vanuit het hoogste onderwijskundige niveau te bekijken haal ik de visie documenten van de nieuwe curricula voor natuurkunde en NL&T aan. Er wordt in de visiedocumenten van de vernieuwde vakken nadrukkelijk gesteld dat er in het onderwijs betere aansluiting op modern onderzoek moet zijn. Dit wordt gestimuleerd door het platform beta techniek, waar ook NI en Philips bij zijn aangesloten. Het huidige curriculum fysische informatica is gebaseerd op TTL technologie uit 1980. Ik denk daarom dat een product als LabView niet alleen een uitstekend hulpmiddel is, maar ook dat LabView het curriculum kan herdefiniëren. Gelukkig wordt die ruimte ook geboden. Zowel in NLT als in NiNa is er nadrukkelijk ruimte voor vernieuwing.

## Van LEGO Mindstorms naar LabView via ...

Lego Mindstorms (binnenkort NXT) is een product dat zich richt op kinderen in de leeftijdsgroep van groep 8 tot de derde klas van de middelbare school. Met deze software en bijbehorende microprocessor kun je zelfgebouwde robots te programmeren. De gebruikersinterface van deze software lijkt sterk op die van LabView. De software is gebouwd met LabView. In de vervolgoopleidingen van het VO (HBO's en universiteiten) krijgen de studenten direct in het eerste jaar te maken met het product LabView. Omdat de producten voor de aansluitende leeftijdsgroepen al bestaan, lijkt het dus zeer wel haalbaar om voor de doelgroep (bovenbouw H/V) deze moderne programmeeromgeving beschikbaar te maken. Dit te onderzoeken is het hoofddoel van dit haalbaarheidsonderzoek.

## Vier aspecten

Uitgaande van het huidige curriculum zijn er vier onderwijsaspecten waar LabView vernieuwend kan worden ingezet.

LabView kan een rol spelen in de hele empirische methodologie van de betavakken. Voor de middelbare school komt dat neer op

- Data acquisitie, verwerking en presentatie (experimenteer omgeving)
- Fysische informatica (sturen en regelen, voorbeeld ontwerp een spoorwegovergang etc.)
- Virtuele instrumentatie (vervanging/aanvulling van apparatuur oscilloscoop, functiegenerator etc.)
- Dynamisch modelleren (voorbeeld biologie populatiedynamica vossen en konijnen, scheikunde reactiemechanismen etc.)

Deze aspecten licht ik hieronder nader toe. Ook zijn er nog overwegingen van andere aard die een rol kunnen spelen. Ook die vind je hieronder.

## Ad data acquisitie

Data acquisitie is de belangrijkste van de vier onderwijsgebieden. Een vernieuwde meetomgeving is het meest nodig. Met een hardwarekaart USB 6009 analoog in 48kS/s gelijkwaardige prijs/prestatie met andere producten.

[Hoe zit dat met Ingangs/uitgangsbeveiliging? leerlingproof maken]

[uitwerken: diverse experimentele configuraties; bibliotheek van experimenten; van kookboekproef tot privé lab]

LabView draait op alle gangbare platforms. Er is ook een versie voor PocketPC. Met een PocketPC in je hand meet je door het hele land.

Video meten. Onder deze naam staat het opmeten van foto's (tweedimensionaal) of van video's (een – of tweedimensionaal in de tijd). De formaten en codexen veranderen met de dag. De huidige

software heeft hier grote moeite mee. [meer info over wat LabView hier mee kan (extra licentie nodig?)].

### **Ad fysische informatica**

Dit onderwijs aspect kan door LabView flink worden verbeterd. Voorbeeld:

In het thans meest gebruikte systeembord zit een drukschakelaartje (die werkt als een deurbeldrukker). Ingedrukt is ie waar, maar laat je hem los dan is ie weer onwaar. Het dogma van het fysische informatica dat ik mijn kindertjes ook leer: Moet een signaal waar blijven gebruik dan een geheugencel. Raar natuurlijk, je moet gewoon een toggle schakelaar nemen, dat is ook het geval als je naast de deur op de lichtknop drukt. Het licht blijft dan echt wel aan.

In software die in de loop van de jaren beschikbaar kwam is dit probleem meegenomen. Er zijn drukschakelaars, omschakelaars en keuzeschakelaars bijgeleverd. Pas op het landelijk examen houdt met deze moderniteiten geen rekening.

In LabView kan je een schakelaar definiëren zoals je wil. Gevaar je kan te veel. Oplossing: biedt in een standaard programmabibliotheek de meest gangbare schakelaars aan.

Ik heb een prototype van een spoorwegovergang in LabView in voorbereiding. Ik heb ook een versie met de software versie van het systeembord gemaakt..

### **Ad virtuele instrumentatie**

Alle routines in LabView heten VI's. Onder dit kopje bedoel ik met virtuele instrumenten werkelijk virtuele scopes, toongenerators etc.

De analoog out van de USB 6009 is zeer beperkt (150 kS/s). Niet voldoende om een toongenerator te maken (hoe los je dat op? externe VCO?). Oscilloscoop werkt tot 48 kS/s. Ik zou een echte oscilloscoop in het natuurkunde kabinet achter de hand houden.

Je kunt ook averagers, dataloggers, voltmeters, thermometers maken. Zowel voor demonstratie (lekker groot display) als voor individueel gebruik. Je kunt de experimenten als exe file opslaan en versturen naar collega's (moeten dan wel met de zelfde hardware werken)

Hoe zit het met interfacing naar en van soundcards? Als dat kan is in ieder geval het audio bereik gedekt.

### **Ad dynamisch modelleren**

Deze onderwijsdoelen zijn het makkelijkst uit te leggen aan de hand van een voorbeeldje (radioactief verval). De leerlingen worden in de leerboeken met semi-code geconfronteerd in de vorm startwaarden en rekenregels:

Radioactief verval

Startwaarden

$n=10000$  'aantal kernen  
 $t=0$  'je moet ergens beginnen  
 $\alpha = -0.04$  'vervalconstante  
 $dt=0.1$  sec

Rekenregels

$dn = \alpha * dt$   
 $n = n + dn$   
 $t = t + dt$

Samen met Gerard Schoep (Univ Leiden) ben ik nu bezig om dit simpele modelletje wat uit te werken en hieruit een algemene vorm te destilleren die alleen de essentie van het programmeren zichtbaar houdt voor de leerlingen.

### **Intuitieve progammeeromgeving**

LabView werkt met twee schermen. Een frontpanel werkt als user interface. Op het frontpanel vindt de programmabesturing en de in- en uitvoer plaats. Er is een zeer uitgebreide bibliotheek beschikbaar om de in- en uitvoer en de programmabesturing te regelen. Met schuiven, thermometers, naaldaanwijzers, drukknoppen, scoopbeelden, fileboxen en dergelijke op kwijt. In het andere scherm, het block-diagram worden de rekenregels geprogrammeerd. Het is alsof je een apparaat open schroeft en bij de besturing komt. De programmeertaal is strikt grafisch en zeer intuïtief. Amerikanen hebben daar een goed oog voor. Je hoeft als simpele gebruiker geen regel code te schrijven. Bovendien werken de schermen nauw samen. "On the fly" worden je acties op de twee schermen bijgehouden, en als je

zondigt tegen de programmeerregels dan wordt dat direct gesignaleerd. Dynamische, context gevoelige help brengt je dan hopelijk weer op het spoor. De principes van de grafische programmeeromgeving kreeg ik in een uurtje duidelijk aan mijn NT leerlingen. Dit was voor mij een van de belangrijkste argumenten om door te gaan met het onderzoek aan LabView in de klas. Er is echter heel wat werk en expert visie nodig om de programmeeromgeving zodanig aan te passen dat middelbare scholieren met LabView aan de slag kunnen. Dit is de grootste uitdaging in het LabView in de klas project. De oplossing ligt zoals altijd bij grote problemen: in stukken hakken Bij het kopje gefaseerde invoering ga ik er verder op in.

### **Gefaseerde invoering**

Een groot project moet je niet in een keer willen rond willen hebben. Kleine stapjes. Deeloplossingen met uitvoerbare doelen. Dat kan hier op twee manieren.

In de loop van de tijd wordt steeds meer functionaliteit van LabView aangeboden (wel in alle vier de onderwijsgebieden). Bijvoorbeeld eerst kant en klare VI's die gebouwd zijn voor een experimenten (al is het een thermometer uitlezing). Een andere manier om te faseren ligt in de opbouw van het curriculum over de verschillende leerjaren. Eerst in de bovenbouw van het VWO komen de vier onderwijsdoelenvolledig aan de orde. Ook in de onderbouw kan al voorzichtig kennis gemaakt worden, zonder de leerlingen en hun leraren overmatig te confronteren met de 'onderwateractiviteiten'. Bijvoorbeeld in de vorm van gecompileerde kookboekexperimenten die alleen via het frontpaneel kunnen worden bestuurd. Het lijkt me geen enkel probleem onderwijskundig verantwoorde tussenoplossingen te verzinnen voor verschillende leerlagen en verschillende cognitieve niveau's. Zo zou je in de onderbouw enige gecompileerde [met runtime engine?] experimenten kunnen opnemen. Voorbeelden te over

- Een digitale thermometer (opwarmen van smeltend ijs)
- Kennismaken met de oscilloscoop
- Kennismaken met toongenerator
- data verzamelen en een tabel en verwerken in een grafiek

In een hogere klas kan je een experiment geheel door de computer laten uitvoeren, en meer functionaliteit invoeren Bijvoorbeeld file I/O. export naar Excell, interfacing met een ELO.

Nog wat sellingpoints waarmee zich LabView onderscheidt van andere producten

- Professionele software dus ondersteuning voor systeembeheerders
- Verantwoorde netwerkinstallatie
- Open interface (geen eigen stekertjes, voor sensoren en geen beschermde protocollen)
- Actieve usergroups
- public repositories Vis ontwikkelen voor en door anderen
- Interfacing met ELO's
- Het project zal aardig passen in Het nieuwe betavak nu bekend als Natuur leven en technologie (NLT).
- Financiële overweging: inzetbaar bij vele vakken, algemeen ICT middel

### **Kritische noot**

In je enthousiasme schrijf je een de zonnige kant van een project zo op. Kritische noten zijn ook Het programmeren in LabView is conceptueel makkelijk, maar dat is bedrieglijk. Ik loop tegen problemen aan waar anderen zeker ook tegenaan zouden lopen. Deze problemen zijn wellicht op te lossen door de programmeeromgeving te beperken of door het aanpassen van default instellingen in ExpressVI's. Bijvoorbeeld: Bij het maken van een grafiek een simulatie meten de leerlingen niet geconfronteerd worden met ingewikkelde databussen. De oplossing ligt in het schrijven van een bib met de juiste subVI's. \*\*

De universiteit van Leiden helpt me met een licentie en hardware. Met Gerard Schoep van het eerstejaarspracticum in Leiden werk ik enige prototypes uit. Wellicht dat we hieruit wat algemeenheden kunnen synthetiseren. Ik denk dat we de tijd moeten gebruiken om onderwijsvoorbeelden te bouwen, en wel zo algemeen mogelijk Er is daarna expert kennis nodig is om dit goed op te lossen.

### **Plan**

Het eerste doel is een haalbaarheidsonderzoek. Voor een aantal kritisch welgezinden zal ik een bijeenkomst beleggen. (okt 2006) Uit deze bijeenkomst vloeit een haalbaarheidsanalyse voort. Deze

bijeenkomst moet goed voorbereid worden. Daarvoor wil ik enige voorbeelden hebben werken. Het liefst een simpel voorbeeld van elk van de onderwijsdoelen.

Kritische noten t.a.v. implementatie, leercurven (twee maal: van leerlingen en docenten) en het noemen van andere voetangels en klemmen zijn dan van harte welkom. Voor deze bijeenkomst wil ik ook collega's uitnodigen die zich voor dit project als ontwikkel/proefschoon kunnen aanmelden.

Op de bijeenkomst kunnen verder aanwezig zijn

- vertegenwoordiger van NI
- Arjen Gerve (Philips)
- Gerard Schoep (univ Leiden)
- vertegenwoordiger van NLT, Nieuwe Natuurkunde (Scheikunde Biologie, wiskunde)
- vertegenwoordiger van vakorganisatie NVON
- Subsidiegevers

Henk Buisman 13 juni 2006