University of Amsterdam
Programming Research Group

UvA Budget Allocatie Model

J.A. Bergstra
S. Nolst Trenite
M.B. van der Zwaag

Report PRG0805 March 2008

J.A. Bergstra

Programming Research Group
Faculty of Science
University of Amsterdam

Kruislaan 403
1098 SJ Amsterdam
The Netherlands

tel. +31 20 525.7591
e-mail: janb@science.uva.nl

S. Nolst Trenite

Faculty of Science
University of Amsterdam

e-mail: sanne@science.uva.nl

M.B. van der Zwaag

Programming Research Group
Faculty of Science
University of Amsterdam

Kruislaan 403
1098 SJ Amsterdam
The Netherlands

tel. +31 20 525.7584
e-mail: mbz@science.uva.nl

Programming Research Group Electronic Report Series

UvA Budget Allocatie Model

Jan Bergstra Sanne Nolst Trenité
Mark van der Zwaag

Faculteit der Natuurwetenschappen, Wiskunde en Informatica
Universiteit van Amsterdam
{janb, sanne, mbz}@science.uva.nl

Inhoudsopgave

1 Introductie

We presenteren een Budget Allocatie Model (BAM) voor de Universiteit van Amsterdam (UvA). Het model definieert de jaarlijkse budgetallocatie aan de faculteiten van de UvA. We hebben ons gebaseerd op beschikbare documenten (waaronder de *Uitwerking voorstel nieuw allocatiemodel* v2.2, begrotingen en kaderbrieven) en correspondentie met de afdeling Concern Control. Opmerkingen ten aanzien van onze interpretatie van deze bronnen dan wel van de gehanteerde praktijk zijn zeer welkom.

In §2 wordt puntsgewijs de omvang van de budgetallocatie aan de faculteiten gedefinieerd. De definitie wordt gegeven in termen van een aantal parameters; deze parameters bepalen de omvang van de allocatie, maar hun waardering valt buiten het bereik van het BAM. We onderscheiden twee soorten parameters:

- Parameters waarvan de waarde onttrokken wordt aan de universitaire informatiesystemen. Hieronder vallen bijvoorbeeld aantallen uitgereikte diploma's, instroomcijfers of gegenereerd inkomen uit contract onderwijs.

- Parameters waarvan de waardering afhangt van beleidsbepalingen, zoals bijvoorbeeld de hoogte van de onderzoeksopslag of de hoogte van de vergoeding per uitgereikt diploma.

De gebruikte parameters worden in §3 in meer detail besproken.

De gestructureerde definitie in §2 laat toe dat we het model formaliseren in wiskundige regels. In §4 presenteren we een formalisering van het BAM in de Tuplix Calculus waarin we de puntsgewijze definitie van §2 volgen en waarin de parameters van het model corresponderen met vrije variabelen.

2 Budget Allocatie Model

2.1. De budgetallocatie aan een faculteit van de UvA bestaat uit twee delen:

(a) Een vastgesteld deel, zie verder Punt 2.2.

(b) Een deel waarvan de omvang afhangt van informatie onttrokken aan de universitaire informatiesystemen, zie verder Punt 2.3.

2.2. Het vastgestelde deel van de allocatie bestaat uit de volgende delen:

(a) Een toegewezen allocatie op grond van nationale schema's. Op dit moment ontvangt alleen de Faculteit der Geesteswetenschappen een dergelijke allocatie, nl. een financiering van onderwijs in kleine talen.

(b) Een additionele allocatie op grond van buitengewone omstandigheden. Er wordt naar gestreefd deze zo min mogelijk toe te kennen. De toekenning wordt vastgelegd in convenanten.

(c) Een strategische allocatie ter versterking van onderwijs en onderzoek dat als speerpunt deel uitmaakt van de universitaire missie.

(d) Een tegemoetkoming in buitengewone kosten aan huisvesting (het huisvestingsexcedent).

Voor elk van deze vier delen wordt de omvang gegeven als parameter van het model.

2.3. Het onder Punt 2.1b genoemde deel van de allocatie bestaat uit een begrenzing van het verder in Punt 2.4 besproken deel.

De begrensde waarde voor de huidige periode is gelijk aan het onder Punt 2.4 besproken deel indien deze waarde niet meer dan een gegeven percentage afwijkt van de begrensde waarde zoals berekend voor de voorgaande periode. Als de afwijking groter is, dan is de begrensde waarde gelijk aan de waarde met de maximale afwijking.

Dit deel is afhankelijk van twee parameters: het gehanteerde afwijkingspercentage en de omvang van de toegewezen, begrensde allocatie in de voorgaande periode, zie verder (3.1).

2.4. De allocatie op basis van informatie onttrokken aan de universitaire informatiesystemen bestaat uit twee delen:

(a) Een onderwijsdeel, zie verder Punt 2.5.

(b) Een onderzoeksdeel, zie verder Punt 2.8.

2.5. Het onderwijsdeel bestaat uit het deel besproken onder Punt 2.6, verhoogd met de onderzoeksopslag, zie verder (3.2).

2.6. Het onderwijsdeel vóór verhoging met de onderzoeksopslag bestaat uit het deel besproken onder Punt 2.7 vermenigvuldigd met het van de faculteit afhankelijke bekostigingsniveau, zie verder (3.3).

2.7. Het onderwijsdeel vóór verhoging ten gevolge van bekostigingsniveau en onderzoeksopslag (zie Punten 2.5 en 2.6) bestaat uit drie delen:

(a) Het aantal door de faculteit toegekende EC's vermenigvuldigd met de vastgestelde vergoeding per EC, zie verder (3.4).

(b) Het aantal door de faculteit toegekende masterdiploma's vermenigvuldigd met de vastgestelde vergoeding per masterdiploma, zie verder (3.5).

(c) De omvang van de instroom van eerstejaars aan de faculteit vermenigvuldigd met de vastgestelde vergoeding per instromer, zie verder (3.6).

2.8. Het onderzoeksdeel bestaat uit de volgende delen:

(a) Het aantal door de faculteit gerealiseerde promoties, achtereenvolgens vermenigvuldigd met de vastgestelde vergoeding per promotie, zie verder (3.7), en het faculteits-afhankelijke bekostigingsniveau.

(b) Een onderzoeksopslag over de toegewezen allocatie (2.2a).

(c) Een onderzoeksopslag over het inkomen van de faculteit uit contract onderwijs, zie verder (3.8).

(d) Een tegemoetkoming in de 'matching' kosten van extern gefinancieerd onderzoek, zie verder (3.9).

3 Parameters

We bespreken de parameters die de omvang van de in §2 gedefinieerde allocatie bepalen. De waardering van de parameters hangt af van de periode waarvoor de allocatie bepaald wordt, en van de faculteit waarvoor de allocatie bepaald wordt.

3.1. Begrenzing van deel (2.3) van de allocatie.

(a) Begrenzingspercentage.

De maximale afwijking ten opzichte van de toekenning van dit deel in de voorgaande periode is op dit moment gesteld op 3 procent. Deze waarde wordt bepaald door het CvB en geldt voor alle faculteiten.

(b) Allocatiedeel (2.3) zoals berekend en toegewezen in de voorgaande periode.

3.2. Onderzoeksopslag.

Voor alle faculteiten wordt onderwijsdeel (2.6) van de allocatie verhoogd met een opslag van 50 procent ten behoeve van onderzoek. De onderzoeksopslag wordt daarnaast toegepast in het onderzoeksdeel van de allocatie (delen (2.8b) en (2.8c)).

3.3. Bekostigingsniveau.

Onderwijsdeel (2.7) van de allocatie wordt vermenigvuldigd met het van de faculteit afhankelijke bekostigingsniveau. Binnen de UvA (met uitzondering van de faculteiten Geneeskunde en Tandheelkunde) varieert deze waarde van 1.0 tot 1.5.

3.4. EC parameters.

(a) Aantal EC's.

Gebruikt wordt het aantal EC's toegekend door een faculteit in het laatste voorgaande, afgesloten academisch jaar. De waarde wordt bepaald door het Bureau Bestuurlijke Informatie onder verantwoording van het CvB op basis van gegevens onttrokken aan informatiesysteem ISIS.

(b) EC-waarde.

De waarde die wordt gekoppeld aan de toekenning van een enkele EC wordt jaarlijks bepaald door de afdeling Concern Control onder verantwoording van het CvB. De waarde geldt voor alle faculteiten.

In 2007 bedroeg de EC-waarde €95.50.

3.5. Masterdiploma parameters.

(a) Aantal masterdiploma's.

Gebruikt wordt het aantal masterdiploma's toegekend door een faculteit in het laatste voorgaande, afgesloten academisch jaar. De waarde wordt bepaald door het Bureau Bestuurlijke Informatie onder verantwoording van het CvB op basis van gegevens onttrokken aan informatiesysteem ISIS.

(b) Masterdipoma-waarde.

De waarde die wordt gekoppeld aan de toekenning van een enkele EC wordt jaarlijks bepaald door de afdeling Concern Control onder verantwoording van het CvB. De waarde geldt voor alle faculteiten.

In 2007 bedroeg deze waarde €3000.

3.6. Instroom parameters.

(a) Aantal eerstejaars instromers.

Het aantal studenten dat zich voor de eerste keer als bachelor inschrijft aan de UvA en aan de gegeven faculteit in het bijzonder.

De gebruikte waarde betreft de instroom in het voorgaande kalenderjaar en wordt bepaald door het Bureau Bestuurlijke Informatie onder verantwoording van het CvB op basis van gegevens onttrokken aan het informatiesysteem ISIS.

(b) Instroom-waarde.

De waarde per eerstejaars instromer. Deze waarde wordt jaarlijks bepaald door de afdeling Concern Control onder verantwoording van het CvB en is van toepassing op alle faculteiten.

De waarde in 2007 bedroeg €750.

3.7. Promotie parameters.

(a) Aantal promoties.

Deze waarde betreft het aantal gerealiseerde promoties binnen de gegeven faculteit in het voorgaande boekjaar. De waarde wordt bepaald door het Bureau Bestuurlijke Informatie onder verantwoording van het CvB.

(b) Promotie-waarde.

De waarde gekoppeld aan een gerealiseerde promotie. Er is één basisbedrag dat geldt voor alle faculteiten, dat wordt bepaald door de afdeling Concern Control onder verantwoording van het CvB.

3.8. Inkomsten uit contract onderwijs.

Zoals gedefinieerd in Punt 2.8c, ontvangt elke faculteit een onderzoeksopslag over haar inkomsten uit contract onderwijs. De waarde die hiervoor gebruikt wordt is op dit moment vastgelegd als het gemiddelde inkomen van de faculteit per jaar uit contract onderwijs over de jaren 2003–2005.

3.9. Tegemoetkoming matching extern onderzoeksgeld.

Elke faculteit ontvangt een bedrag dat gebaseerd is op de externe financiering van onderzoek (2e en 3e geldstroom) aan dezelfde faculteit. De gebruikte waarde is bepaald als het gemiddelde per jaar aan totale tegemoetkomingen in de matchingkosten[1] berekend over de jaren 2003–2005.

De omvang van de ontvangen externe financiering door faculteiten en de gehanteerde matchingpercentages worden vastgesteld door het CvB, ondersteund door de afdeling Concern Control.

4 Formalisering

We formaliseren het UvA BAM in Tuplix Calculus. We refereren naar [1] voor de syntax en semantiek van Tuplix Calculus. De formalisering van budgetten is onderwerp van [2]; dit artikel bevat bovendien als voorbeeld een uitgebreide toepassing van de Tuplix Calculus.

Zij *Fac* een eindige verzameling faculteitsnamen. Voor de parameters van het model introduceren we variabelen die vrij gebruikt worden in de formalisering. We indexeren deze variabelen met faculteitsnaam $F \in Fac$ wanneer de waarde afhankelijk is van faculteit F. Een opsomming van de variabelen wordt gegeven in Tabel 1. Zie §3 voor een verdere uitleg van de parameters.

[1] Deze tegemoetkomingen voor de 2e en 3e geldstroom zijn gedefinieerd als 70 resp. 30 procent van de ontvangen projectfinanciering.

Variabele	Omschrijving	Zie
bp	begrenzingspercentage	(3.1a)
bavp_F	begrensde allocatie voorgaande periode	(3.1b)
oo	onderzoeksopslag	(3.2)
bn_F	bekostigingsniveau	(3.3)
$\#\mathsf{ec}_F$	aantal EC's	(3.4a)
ec	EC waarde	(3.4b)
$\#\mathsf{md}_F$	aantal masterdiploma's	(3.5a)
md	waarde masterdiploma	(3.5b)
$\#\mathsf{instr}_F$	aantal eerstejaars instromers	(3.6a)
instr	waarde per instromer	(3.6b)
$\#\mathsf{phd}_F$	aantal promoties	(3.7a)
phd	waarde per promotie	(3.7b)
ct_F	omvang contract onderwijs	(3.8)
ma_F	omvang matching onderzoeksgelden	(3.9)
toe_F	toegewezen allocatie	(2.2a)
add_F	additionele allocatie	(2.2b)
stra_F	strategische allocatie	(2.2c)
hve_F	huisvestingsexcedent	(2.2d)

Tabel 1: Variabelen voor de parameters

We gebruiken (gebonden) variabelen voor de delen van de allocatie die worden gedefinieerd onder Punt 2.1–2.8 in §2. We gebruiken variabele x^{p} voor het deel van de allocatie gedefinieerd onder Punt p in §2.

De definitie van het BAM in §2 voor faculteit F vatten we nu in de volgende conditie in Zero-Test Logic (zie [1]):

$$
\begin{aligned}
\phi_F = \\
& x^{2.1} = x^{2.2} + x^{2.4} \wedge \\
& x^{2.2} = \mathsf{toe}_F + \mathsf{add}_F + \mathsf{stra}_F + \mathsf{hve}_F \wedge \\
& x^{2.4} = x^{2.5} + x^{2.8} \wedge \\
& x^{2.5} = (1 + \mathsf{oo}) \cdot x^{2.6} \wedge \\
& x^{2.6} = \mathsf{bn}_F \cdot x^{2.7} \wedge \\
& x^{2.7} = \mathsf{ec} \cdot \#\mathsf{ec}_F + \mathsf{md} \cdot \#\mathsf{md}_F + \mathsf{instr} \cdot \#\mathsf{instr}_F \wedge \\
& x^{2.8} = \mathsf{bn}_F \cdot \mathsf{phd} \cdot \#\mathsf{phd}_F + \mathsf{ma}_F + \mathsf{oo} \cdot (\mathsf{toe}_F + \mathsf{ct}_F)
\end{aligned}
$$

We hebben de begrenzing (Punt 2.3) voor dit moment achterwege gelaten, omdat formalisering hiervan een extra aanname op het datatype vereist; we komen hier later in deze paragraaf op terug.

We gebruiken attribuutnaam a_F voor betalingen aan faculteit F. Verder is $\bar{x}$ een aanduiding voor de lokale variabelen x^{p} die gebruikt worden in de definitie van de condities ϕ_F. Definieer nu, voor $F \in Fac$,

$$P_F = \sum_{\bar{x}}(\gamma(\phi_F) \oplus a_F(x^{2.1})),$$

voor de institutionele budgetallocatie aan F. Dit maakt deel uit van een institutioneel budget. We definiëren verder facultaire budgetten

$$Q_F = \sum_x(a_F(-x) \oplus R_F),$$

waarbij R_F een aanduiding is voor het interne deel van het facultaire budget van F.

Neem nu verzameling

$$H = \{a_F \mid F \in Fac\}.$$

van attributen en neem aan dat de interne facultaire budgetten R_F de attributen in H niet gebruiken. Nu volgt dat

$$\partial_H \left(\bigoplus_{F \in Fac}(P_F \oplus Q_F)\right)$$

het BAM afdwingt op de facultaire budgetten.

Formalisering van de begrenzing op de allocatie.

Voor het formaliseren van de begrenzing op de allocatie, zoals beschreven onder Punt 2.3, is het nodig om naast vergelijkingen ook ongelijkheden uit te drukken in de conditie ϕ_F. In het standaard datatype voor de Tuplix Calculus is dit niet direct mogelijk. Het kan wel als we het datatype eerst uitbreiden met een absoluut-operator. Laten we dit doen: neem de absoluut-operator $|\_|$ (met $|p| = p$ als $p \geq 0$, and $|p| = -p$ anders). Dan drukt de vergelijking

$$|q - p| = q - p$$

de ongelijkheid $p \leq q$ uit.

De volgende conditie in Zero-Test Logic is nu een uitdrukking van de begrenzing (2.3) op de allocatie:

$$\begin{aligned}
\psi_F = \\
&(x^{2.4} \geq (1 + \mathsf{bp}) \cdot \mathsf{bavp}_F \wedge x^{2.3} = (1 + \mathsf{bp}) \cdot \mathsf{bavp}_F) \vee \\
&(x^{2.4} \leq (1 - \mathsf{bp}) \cdot \mathsf{bavp}_F \wedge x^{2.3} = (1 - \mathsf{bp}) \cdot \mathsf{bavp}_F) \vee \\
&(|x^{2.4} - \mathsf{bavp}_F| < \mathsf{bp} \cdot \mathsf{bavp}_F \wedge x^{2.3} = x^{2.4})
\end{aligned}$$

De volledige conditie ϕ_F wordt nu:

$$\begin{aligned}\phi_F = \psi_F \wedge &\\ & x^{2.1} = x^{2.2} + x^{2.3} \wedge\\ & x^{2.2} = \mathsf{toe}_F + \mathsf{add}_F + \mathsf{stra}_F + \mathsf{hve}_F \wedge\\ & x^{2.4} = x^{2.5} + x^{2.8} \wedge\\ & x^{2.5} = (1 + \mathsf{oo}) \cdot x^{2.6} \wedge\\ & x^{2.6} = \mathsf{bn}_F \cdot x^{2.7} \wedge\\ & x^{2.7} = \mathsf{ec} \cdot \#\mathsf{ec}_F + \mathsf{md} \cdot \#\mathsf{md}_F + \mathsf{instr} \cdot \#\mathsf{instr}_F \wedge\\ & x^{2.8} = \mathsf{bn}_F \cdot \mathsf{phd} \cdot \#\mathsf{phd}_F + \mathsf{ma}_F + \mathsf{oo} \cdot (\mathsf{toe}_F + \mathsf{ct}_F)\end{aligned}$$

De verdere formalisering blijft ongewijzigd.

Referenties

[1] J.A. Bergstra, A. Ponse and M.B. van der Zwaag. Tuplix Calculus. arXiv.org, arXiv:0712.3423v1 [cs.LO], 2007.

[2] J.A. Bergstra, S. Nolst Trenité and M.B. van der Zwaag. Towards a formalization of budgets. arXiv.org, arXiv:0802.3617v1 [cs.LO], 2008.

Electronic Reports Series of the Programming Research Group

Within this series the following reports appeared.

[PRG0804] J.A. Bergstra and C.A. Middelburg, *Thread Algebra for Sequential Poly-Threading,* Programming Research Group - University of Amsterdam, 2008.

[PRG0803] J.A. Bergstra and C.A. Middelburg, *Thread Extraction for Polyadic Instruction Sequences,* Programming Research Group - University of Amsterdam, 2008.

[PRG0802] A. Barros and T. Hou, *A Constructive Version of AIP Revisited,* Programming Research Group - University of Amsterdam, 2008.

[PRG0801] J.A. Bergstra and C.A. Middelburg, *Programming an Interpreter Using Molecular Dynamics,* Programming Research Group - University of Amsterdam, 2008.

[PRG0713] J.A. Bergstra, A. Ponse, and M.B. van der Zwaag, *Tuplix Calculus,* Programming Research Group - University of Amsterdam, 2007.

[PRG0712] J.A. Bergstra, S. Nolst Trenite, and M.B. van der Zwaag, *Towards a Formalization of Budgets,* Programming Research Group - University of Amsterdam, 2007.

[PRG0711] J.A. Bergstra and C.A. Middelburg, *Program Algebra with a Jump-Shift Instruction,* Programming Research Group - University of Amsterdam, 2007.

[PRG0710] J.A. Bergstra and C.A. Middelburg, *Instruction Sequences with Dynamically Instantiated Instructions,* Programming Research Group - University of Amsterdam, 2007.

[PRG0709] J.A. Bergstra and C.A. Middelburg, *Instruction Sequences with Indirect Jumps,* Programming Research Group - University of Amsterdam, 2007.

[PRG0708] B. Diertens, *Software (Re-)Engineering with PSF III: an IDE for PSF,* Programming Research Group - University of Amsterdam, 2007.

[PRG0707] J.A. Bergstra and C.A. Middelburg, *An Interface Group for Process Components,* Programming Research Group - University of Amsterdam, 2007.

[PRG0706] J.A. Bergstra, Y. Hirschfeld, and J.V. Tucker, *Skew Meadows,* Programming Research Group - University of Amsterdam, 2007.

[PRG0705] J.A. Bergstra, Y. Hirschfeld, and J.V. Tucker, *Meadows,* Programming Research Group - University of Amsterdam, 2007.

[PRG0704] J.A. Bergstra and C.A. Middelburg, *Machine Structure Oriented Control Code Logic (Extended Version),* Programming Research Group - University of Amsterdam, 2007.

[PRG0703] J.A. Bergstra and C.A. Middelburg, *On the Operating Unit Size of Load/Store Architectures,* Programming Research Group - University of Amsterdam, 2007.

[PRG0702] J.A. Bergstra and A. Ponse, *Interface Groups and Financial Transfer Architectures,* Programming Research Group - University of Amsterdam, 2007.

[PRG0701] J.A. Bergstra, I. Bethke, and M. Burgess, *A Process Algebra Based Framework for Promise Theory,* Programming Research Group - University of Amsterdam, 2007.

[PRG0610] J.A. Bergstra and C.A. Middelburg, *Parallel Processes with Implicit Computational Capital,* Programming Research Group - University of Amsterdam, 2006.

[PRG0609] B. Diertens, *Software (Re-)Engineering with PSF II: from architecture to implementation,* Programming Research Group - University of Amsterdam, 2006.

[PRG0608] A. Ponse and M.B. van der Zwaag, *Risk Assessment for One-Counter Threads,* Programming Research Group - University of Amsterdam, 2006.

[PRG0607] J.A. Bergstra and C.A. Middelburg, *Synchronous Cooperation for Explicit Multi-Threading,* Programming Research Group - University of Amsterdam, 2006.

[PRG0606] J.A. Bergstra and M. Burgess, *Local and Global Trust Based on the Concept of Promises,* Programming Research Group - University of Amsterdam, 2006.

[PRG0605] J.A. Bergstra and J.V. Tucker, *Division Safe Calculation in Totalised Fields,* Programming Research Group - University of Amsterdam, 2006.

[PRG0604] J.A. Bergstra and A. Ponse, *Projection Semantics for Rigid Loops,* Programming Research Group - University of Amsterdam, 2006.

[PRG0603] J.A. Bergstra and I. Bethke, *Predictable and Reliable Program Code: Virtual Machine-based Projection Semantics (submitted for inclusion in the Handbook of Network and Systems Administration),* Programming Research Group - University of Amsterdam, 2006.

[PRG0602] J.A. Bergstra and A. Ponse, *Program Algebra with Repeat Instruction,* Programming Research Group - University of Amsterdam, 2006.

[PRG0601] J.A. Bergstra and A. Ponse, *Interface Groups for Analytic Execution Architectures,* Programming Research Group - University of Amsterdam, 2006.

[PRG0505] B. Diertens, *Software (Re-)Engineering with PSF,* Programming Research Group - University of Amsterdam, 2005.

[PRG0504] P.H. Rodenburg, *Piecewise Initial Algebra Semantics,* Programming Research Group - University of Amsterdam, 2005.

[PRG0503] T.D. Vu, *Metric Denotational Semantics for BPPA,* Programming Research Group - University of Amsterdam, 2005.

[PRG0502] J.A. Bergstra, I. Bethke, and A. Ponse, *Decision Problems for Pushdown Threads,* Programming Research Group - University of Amsterdam, 2005.

[PRG0501] J.A. Bergstra and A. Ponse, *A Bypass of Cohen's Impossibility Result,* Programming Research Group - University of Amsterdam, 2005.

[PRG0405] J.A. Bergstra and I. Bethke, *An Upper Bound for the Equational Specification of Finite State Services,* Programming Research Group - University of Amsterdam, 2004.

[PRG0404] J.A. Bergstra and C.A. Middelburg, *Thread Algebra for Strategic Interleaving,* Programming Research Group - University of Amsterdam, 2004.

[PRG0403] B. Diertens, *A Compiler-projection from PGLEc.MSPio to Parrot,* Programming Research Group - University of Amsterdam, 2004.

[PRG0402] J.A. Bergstra and I. Bethke, *Linear Projective Program Syntax,* Programming Research Group - University of Amsterdam, 2004.

[PRG0401] B. Diertens, *Molecular Scripting Primitives,* Programming Research Group - University of Amsterdam, 2004.

[PRG0302] B. Diertens, *A Toolset for PGA,* Programming Research Group - University of Amsterdam, 2003.

[PRG0301] J.A. Bergstra and P. Walters, *Projection Semantics for Multi-File Programs,* Programming Research Group - University of Amsterdam, 2003.

[PRG0201] I. Bethke and P. Walters, *Molecule-oriented Java Programs for Cyclic Sequences,* Programming Research Group - University of Amsterdam, 2002.

The above reports and more are available through the website: www.science.uva.nl/research/prog/

Electronic Report Series

Programming Research Group
Faculty of Science
University of Amsterdam

Kruislaan 403
1098 SJ Amsterdam
the Netherlands

www.science.uva.nl/research/prog/