P

Pandora 168
parallel clause composition 172
parallel conjunction 172
parallel execution 1
parallelism 4, 166
parameter
  actual 10
  formal 10, 46
PARLOG 7, 61, 172
partial execution 76
permanent variable 98
Polka 120
port 33
port reduction rules 133
ports 121, 131
  as constraints 132
  closing 131
  implementation 135
  opening 131
  recognising 131
  sending to 131
PRAM 138
procedure-based language 6
proceed, with and-tasks 91
process 25
process communication 25
process network 28
process-based language 6
producer 25
program 46
program atom 45
program atom execution rule 69
program atom rule 50
program atom statement 10
program counter register 86
program name 45
programming paradigms 5, 24
Prolog 7, 36, 153-169
promote after splitting rule 75
promote and-node 90
promote task 65, 84
promotion execution rule 71, 72
promotion rule 50
propagation of constraints 38
prune and-nodes 91
pruning 159
  noisy 159
  quiet 159

Q

quiescent goal 54
quiet and-node 87
quiet constraint 47
quiet cut rule 57
quiet pruning 159

R

RAM 139
rational tree constraints,
  simplification of 53
rational trees 47
  stability for 55
  suspending and waking on 67
read-only variable 169
reader of medium 123
receive on medium 123
reconciliation, cc combinator 185
records 48
reduction rule
  clausewise 154
  definitionwise 155
  local forking 169
references 81
register allocation 103
registers 86
Relational Language 169
relational programming 36
relative simplification 53
residuation 168
restore insertion point task 84
resume task 84
rules
  computation 49-58
  execution 69-76
  port reduction 133

S

satisfiable constraint 47
select candidate 92
semantics 3
send
  on medium 123
  to port 131
sequencing 145, 175
sequential clause composition 172
sequential conjunction 172
sharing scheme, general 113

sharing scheme, simple 115
sho-en, in KL1 172
short-circuiting 126, 141, 145, 174
side effects 143, 163
simplification
  of rational tree constraints 53
  relative 53
simplification mode 49
single worker execution 64
single-solution group, cc combinator 185
SLD-resolution 155
soft cut 57
solved and-box 48
solved and-node 87
soundness
  of answers 59
  of failure 59
stable and-box detection rule 72
stable and-node 82
stable goal 51, 54
stable local store 19
statement 10, 46
  aggregate 46
  bagof 20
  choice 46
  committed choice 14, 46
  composition 10, 46
  conditional choice 10, 17, 46
  constraint atom 10, 45
  hiding 11, 46
  nondeterminate choice 15, 46
  program atom 10, 45
statement task 65
statements 10-23
  summary 23
static store 85
store 8
Strand 7
stream merger 124
stream distributor, n-ary 139
streams 124
stuck configuration 56
subgoal 49
subgoal rule 49
substitution 47
success continuation 158
suspend and-node 89
suspend, binding in GHC 170
suspend, on variable 65
suspended call 107
suspension 65, 83
suspension, ignoring variables for 66
symbol 83
synchronisation idioms 136
syntax 3
  character set 23
  clause 21
  conventions 11, 17, 21
  underscore symbol 23
synthesize, cc combinator 185

T

task 65
  and-box 65
  choice splitting 65
  clause 65
  impose 65
  install 65
  or-box 65
  promote 65
  statement 65
  wake 65
task execution rules 65
task list 65
tasks 84
  and 65
  choice 65
tell 8
temporary variable 98
terminal configuration 56
terminal execution state 76
then operator 17, 23
threading 4
trail entry 84
trail stack 85
transducer 26
transition system, goal 49
transparent message-distribution 126
tree 83
tree constraints 47
tree constructor 83
tree-constraint instructions 96
true 10, 47
tuple space 146

U

underscore symbol 23
unification 47
unification instructions 96
unification mode register 86
unify trees 88
unit execution rule 73

unit goal 48
unit operation 46
unit rule 52
unit statement 52

V

variable 8, 9, 45, 83
    external 64
    local 64
Vulcan 120

W

wait operator 15
wake 90
wake task 65, 84
wake, on variable 65
wake-up rule 75
wake-up stack 85
well-formed goal 55
woken-up rule 75
worker 64
writer of medium 123

X

x-registers 86
    assigning 103

Y

y-registers 83, 86

Computing Science Department, Uppsala University
Uppsala Theses in Computing Science (Doctoral Dissertations)

03. Göran Hagert, *Logic Modeling of Conceptual Structures: Steps Towards a Computational Theory of Reasoning*, 1986
09. Jonas Barklund, *Parallel Unification*, 1990
10. Håkan Millroth, *Reforming Compilation of Logic Programs*, 1990
15. Andreas Hamfelt, *Metalogic Representation of Multilayered Knowledge*, 1992
19. Sverker Janson, *AKL—A Multiparadigm Programming Language*, 1994

Swedish Institute of Computer Science
SICS Dissertation Series

01. Bogumil Hausman, *Pruning and Speculative Work in OR-parallel PROLOG*, 1990
02. Mats Carlsson, *Design and Implementation of an OR-parallel Prolog Engine*, 1990
03. Nabiel A. Elshiewy, *Robust Coordinated Reactive Computing in SANDRA*, 1990
04. Dan Sahlin, *An Automatic Partial Evaluator for Full Prolog*, 1991
05. Hans A. Hansson, *Time and Probability in Formal Design of Distributed Systems*, 1991
06. Peter Sjödin, *From LOTOS Specifications to Distributed Implementations*, 1991
07. Roland Karlsson, *A High Performance OR-parallel Prolog System*, 1991
08. Erik Hagersten, *Toward Scalable Cache Only Memory Architectures*, 1991
09. Lars-Henrik Eriksson, *Finitary Partial Inductive Definitions and General Logic*, 1993
10. Mats Björkman, *Architectures for High Performance Communication*, 1993
11. Steven Pink, *Measurement, Implementation and Optimization of Internet Protocols*, 1993
12. Martin Aronsson, *GCLA: The Design, Use and Implementation of a Program Development System*, 1993
13. Christer Samuelsson, *Fast Natural-Language Parsing Using Explanation-Based Learning*, 1994
14. Sverker Janson, *AKL—A Multiparadigm Programming Language*, 1994

AKL
A Multiparadigm Programming Language
Based on a Concurrent Constraint Framework

Printed by Graphic Systems, Stockholm, Sweden, using a camera-ready copy typeset in Adobe Palatino by the author, using a LaserWriter Pro 810, Microsoft Word 5.1, Adobe Illustrator 5.0, and the Macintosh Quadra 700 and Macintosh PowerBook 170 personal computers.