The astra package*

ASTRA Group on Constraint Programming
<http://www.it.uu.se/research/group/astra>

15th May 2012

1 Introduction

This package provides macros used for the ASTRA Group on Constraint Programming's in-house publication style.

2 Usage

Some expressions for use in math environments are defined in section 4.1.

`\Keyword` Keywords in models and algorithms can be typeset with `\Keyword{`⟨*kywd*⟩`}`, and by default are typeset in boldface (**kywd**). For a list of predefined keywords, see section 4.3.

`\newKeyword` New keywords can be defined with `\newKeyword{`⟨*command*⟩`}{`⟨*output*⟩`}`. For example, `\newKeyword{\foo}{bar}` defines the macro `\foo`, which will be typeset as **bar**. If you need to define a keyword that takes arguments, you'll need to use `\newcommand`. For example, if you want the command `\foo2{some}{thing}` to be typeset as "some **bar** thing", you could add to your preamble:

```
\newcommand{\foo2}[2]{#1~\Keyword{bar}~#2}
```

`\Constraint` Constraints typeset using `\Constraint {`⟨*ConstName*⟩`}` look like this: ConstName. For a list of predefined constraints (such as AllDiff, Linear, etc.), see section 4.2.

`\todo` The `\todo` macro inserts a highlighted note if the package was called with the **todo** option. Otherwise, text in `\todo` is ignored. `\todo{Make text redder!}` looks like this:

To do: Make text redder!

*This document corresponds to astra v1.4, dated 2012/05/15.

`\AstraLogo` You can insert the ASTRA logo with the macro `\AstraLogo`. It uses tikz to draw the logo, and takes a single optional parameter: the scale of the drawing. The default is `0.75`.

`\handpoint` `\tick` `\notick` These macros generate some dingbats: ☞, ✓, ✗.

Figure 1: The ASTRA logo at 0.75 scale.

3 Options

todo Text in `\todo{}` macros will be displayed, with an indicator that it is a todo. Without this option, `\todo` text is not printed. To do: Consider using marginpars for this.

noalg Prevents the package from loading the algorithm and algorithmic packages (in case you prefer another package, or just want to control the options).

alg2e Tells astra that you will use algorithm2e for algorithm typesetting. If **noalg** is also passed, then this option only affects the way certain commands are defined; without the **noalg** option, **alg2e** loads algorithm2e with the **boxed** and **noend** options.

nohyperref Prevents the package from loading hyperref.

xetex Lets astra know you are using XeTeX or XeLaTeX as your typesetting engine. The package will then load polyglossia instead of babel, and will define special characters based on unicode numbers instead of using pifont. Default is to use Zapf Dingbats for dingbat characters; if you prefer another unicode symbol font, specify it in your preamble like this:

```
\newfontfamily{\astrading}{Arial Unicode MS}
```

gets, coloneqq Choose to use 'gets' (←) or 'colon equals' (:=) when indicating `\IsAssigned` variable assignment with the `\IsAssigned` macro. Default option is **gets**.

4 Implementation

```
\newif\if@astra@todo
\DeclareOption{todo}{\@astra@todotrue}
\newif\if@astra@xetex
\DeclareOption{xetex}{\@astra@xetextrue}
\newif\if@astra@alg
\newif\if@astra@algiie
\DeclareOption{alg}{\@astra@algtrue}
\DeclareOption{noalg}{\@astra@algfalse}
\DeclareOption{alg2e}{\@astra@algiietrue}
```

```
\newif\if@astra@hyperref
\DeclareOption{hyperref}{\@astra@hyperreftrue}
\DeclareOption{nohyperref}{\@astra@hyperreffalse}
\newif\if@astra@gets
\DeclareOption{gets}{\@astra@getstrue}
\DeclareOption{coloneqq}{\@astra@getsfalse}
\DeclareOption*{\PackageWarning{astra}{Unknown option `\CurrentOption'}}
\ExecuteOptions{alg,hyperref,gets}
\ProcessOptions\relax

\if@astra@alg
\if@astra@algiie
\RequirePackage[boxed,noend]{algorithm2e}
\else
\RequirePackage[boxed]{algorithm}
\RequirePackage[noend]{algorithmic} % drop [noend] if endif, endwhile, etc wanted
\renewcommand{\algorithmiccomment}[1]{\hfill // #1}
\fi
\fi
\RequirePackage{alltt}
\RequirePackage{amsmath}
\RequirePackage{amssymb}
%%\RequirePackage{amsthm} \newtheorem{notation}{Notation}
\if@astra@xetex
\RequirePackage{xkeyval} % omitted dependency in polyglossia as of 2011/09/20
\RequirePackage{polyglossia}
\setdefaultlanguage[variant=uk]{english}
\defaultfontfeatures{Mapping=tex-text}
\newfontinstance\astrading{Zapf Dingbats}
\newcommand{\handpoint}{{\astrading\char"261E}}
\newcommand{\tick}{{\astrading\char"2713}}
\newcommand{\notick}{{\astrading\char"2717}}
\else
\RequirePackage[british]{babel}
%% Fancy characters:
\RequirePackage{pifont}
%%\newcommand{\scissors}{\ding{36}}
%%\newcommand{\phone}{\ding{37}}
%%\newcommand{\aircraft}{\ding{40}}
%%\newcommand{\envelope}{\ding{41}}
\newcommand{\handpoint}{\ding{43}}
%%\newcommand{\victory}{\ding{44}}
%%\newcommand{\handwrite}{\ding{45}}
\newcommand{\tick}{\ding{51}}
\newcommand{\notick}{\ding{55}}
\fi
\RequirePackage{color}
%%\RequirePackage{float}
\RequirePackage{graphicx}
%%\RequirePackage{grffile} % allows spaces in filenames
```

```
\RequirePackage{mathtools} \mathtoolsset{showonlyrefs,showmanualtags}
\RequirePackage{multicol}
\RequirePackage{multirow}
\RequirePackage{rotating}
\RequirePackage{tikz} \usetikzlibrary{trees}
\if@astra@hyperref
\RequirePackage{hyperref} % should always be the last package
\fi
```

`\AstraLogo` Uses tikz to draw the ASTRA logo. Optional parameter is a scaling factor; 0.75 is the default and results in a nice size for most applications. The font will depend on your document defaults (would like to fix this in future versions; suggestions welcome).

```
\tikzstyle{AstraFont}=[font=\huge\bfseries]
\newcommand{\AstraLogo}[1][0.75]{%
\scalebox{#1}{%
\begin{tikzpicture}[fill=blue,draw=white,line width=2pt]
%%\draw [help lines] (0,0) grid (4,5);
\filldraw (0,5) -- (-1,1) -- (0,2.25) -- (1, 1) -- cycle;
\filldraw (-2,3) -- (-.5,3) -- (0,2.75) -- (.5,3) -- (2,3) -- (0,2.25) -- cycle;
\node[text=blue,AstraFont] at (0,.5) {ASTRA};
\end{tikzpicture}%
}%
}
```

`\blue` `\green` `\red` Colours (use sparingly!):

```
\newcommand{\blue}[1]{{\color{blue}#1}}
\newcommand{\green}[1]{{\color{Green}#1}}
\newcommand{\red}[1]{{\color{red}#1}}
```

`\todo` Only visible if the `todo` option is used.

```
\if@astra@todo
\newcommand{\todo}[1]{\red{To do: #1}}
\else
\newcommand{\todo}[1]{}
\fi
```

`\marginpar` Improve the look of the built-in `\marginpar` macro a bit...

```
\let\oldmarginpar\marginpar
\renewcommand\marginpar[1]{\-\oldmarginpar[\raggedleft\footnotesize #1]%
{\raggedright\footnotesize #1}}
```

4.1 Mathematics

```
\newcommand{\Keyword}[1]{\textbf{#1}}
\newcommand{\newKeyword}[2]{\newcommand{#1}{\Keyword{#2}}}
%\if@astra@algiie

%\renewcommand{\newKeyword}[2]{\SetKwData{#1}{#2}}
%\SetDataSty{\textbf}
%\fi
%% Mathematics
\newcommand{\Cardinality}[1]{\left\lvert#1\right\rvert}
%%\newcommand{\Cardinality}[1]{\##1}
\newcommand{\Ceiling}[1]{\left\lceil#1\right\rceil}
\if@astra@algiie
\else
\newKeyword{\Else}{~else~}
\newKeyword{\False}{false}
\newKeyword{\If}{if~}
\fi
\newcommand{\Floor}[1]{\left\lfloor#1\right\rfloor}
\newcommand{\GeqLex}{\geq_\Lex}
\newcommand{\Iff}{\Leftrightarrow}
\newcommand{\IfThenElse}[3]{\If#1\Then#2\Else#3}
\newcommand{\Implies}{\Rightarrow}
\newcommand{\Inter}{\cap}
\newcommand{\LeqLex}{\leq_\Lex}
\newcommand{\Let}[2]{\Keyword{let~}#1\In#2}
\newcommand{\Lex}{\text{lex}}
\newcommand{\LtLex}{<_\Lex}
\newcommand{\Oh}[1]{\mathcal{O}\left(#1\right)}
\newKeyword{\Select}{select}
\newcommand{\Sequence}[1]{\left[#1\right]}
\newcommand{\Set}[1]{\left\{#1\right\}}
\newcommand{\SetComp}[2]{\Set{#1\SuchThat#2}}
\newcommand{\SuchThat}{\mid}
\newKeyword{\Then}{~then~}
\newKeyword{\True}{true}
\newcommand{\Tuple}[1]{\left\langle#1\right\rangle}
\newcommand{\Union}{\cup}
```

4.2 CP

```
%% CP
\newcommand{\AllDiff}{\Constraint{AllDiff}}
\newcommand{\AllDifferent}{\Constraint{AllDifferent}}
\newcommand{\AllEqual}{\Constraint{AllEqual}}
\newcommand{\AtLeast}{\Constraint{AtLeast}}
\newcommand{\AtMost}{\Constraint{AtMost}}
\newcommand{\Constraint}[1]{\textsc{#1}}
\newcommand{\Cumulative}{\Constraint{Cumulative}}
```

```
\newcommand{\Distinct}{\Constraint{Distinct}}
\newcommand{\Domain}[1]{\text{dom}(#1)}
\newcommand{\Element}{\Constraint{Element}}
\newcommand{\GlobalCardinality}{\Constraint{GCC}}
\newcommand{\HigherOrder}[1]{\red{\left[{\color{black}#1}\right]}} % Iverson bracket
\newcommand{\Inequalities}{\Constraint{Inequalities}}
\newcommand{\IntValuePrecedeChain}{\Constraint{IntValuePrecedeChain}}
\newcommand{\Linear}{\Constraint{Linear}}
\newcommand{\NbDistinct}{\Constraint{NbDistinct}}
\newcommand{\Reifies}[2]{#1\Leftrightarrow#2}
\newcommand{\Sum}{\Constraint{Sum}}
\newcommand{\Variables}[1]{\text{var}(#1)}
```

4.3 Modelling & Algorithms

```
%% Modelling \& Algorithms
\if@astra@algiie
\else
\renewcommand{\algorithmicendfor}{\Keyword{end try}}
\renewcommand{\algorithmicforall}{\Keyword{try all}}
\fi
\newKeyword{\Array}{array}
\newKeyword{\Boolean}{boolean}
\newcommand{\Count}[2]{\Keyword{count}(#1)~#2}
\newKeyword{\Constant}{cst~}
\newKeyword{\EndIf}{~end~if}
%%\newcommand{\Endtry}{\Keyword{end try}}
\newcommand{\Endtry}{\ENDFOR}
\newKeyword{\Exists}{exists~}  % text variant: do not use in models!
%%\newcommand{\Exists}{\exists}          % math variant: do not use in models!
\if@astra@algiie
\else
\newKeyword{\ForAll}{for~all~} % text variant
\fi
%%\newcommand{\ForAll}{\forall}          % math variant
\newKeyword{\Function}{function~}
\newcommand{\IfThen}[2]{\If#1\Then#2\EndIf}
\newKeyword{\In}{~in~} % text variant
%%\newcommand{\In}{\in}          % math variant
\newKeyword{\Integer}{int}
\newcommand{\Invariant}[1]{\Keyword{invariant:} #1}
\if@astra@gets
\newcommand{\IsAssigned}{\ensuremath{\gets}}
\else
\newcommand{\IsAssigned}{\ensuremath{\coloneqq}}
\fi
\newKeyword{\Of}{~of~}
\newKeyword{\Maximise}{maximise}
\newKeyword{\Minimise}{minimise}
```

```
\newcommand{\PostCond}[1]{\Keyword{post:} #1}
\newcommand{\PreCond}[1]{\Keyword{pre:} #1}
\newKeyword{\Procedure}{procedure~}
\newKeyword{\SetDecl}{set~}
\newKeyword{\SetOf}{setof}
\newKeyword{\Size}{size}
\newKeyword{\Solve}{solve}
\newcommand{\SUM}[3]{\\Keyword{sum}(#1)~#3}            % text variant, ignores #2
%%\newcommand{\SUM}[3]{\displaystyle\sum_{#1}^{#2} #3} % math variant
%%\newcommand{\Try}[1]{\STATE {\Keyword{try}~#1}}
%%\newcommand{\Tryall}{\Keyword{try all}}
\newcommand{\Tryall}{\FORALL}
\newKeyword{\Var}{var~}
\newcommand{\Variant}[1]{\Keyword{variant:} #1}
```

Change History

v1.0

General: Initial version (based on macros.tex) 1

v1.1

General: Added support for algorithm2e 1

v1.2

General: Removed `stressed` from macros. 1

v1.3

General: Added showmanualtags option to mathtools package. 1, 4

v1.4

General: Fixed the case of a couple alg2e 'set' commands. 1

Temporarily disabling alg2e specific keyword macros 5

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols

\# 101
\& 149
\- 91
\@astra@algfalse . . . 8
\@astra@algiietrue . . 9
\@astra@algtrue 7
\@astra@getsfalse . 15
\@astra@getstrue . . 14
\@astra@hyperreffalse 12
\@astra@hyperreftrue 11
\@astra@todotrue . . . 2
\@astra@xetextrue . . 4
\\ 190
\{ 122
\} 122

A

\aircraft 48
\algorithmiccomment 26
\algorithmicendfor . 152
\algorithmicforall . 153
\AllDiff 130
\AllDifferent 131
\AllEqual 132
\Array 155
\astrading 38–41
\AstraLogo 71
\AtLeast 133
\AtMost 134

B

\begin 74
\bfseries 71
\blue 82

\Boolean 156

C

\cap 114
\Cardinality . . 100, 101
\Ceiling 102
\char 39–41
\coloneqq 178
\color 82–84, 141
\Constant 158
\Constraint 130–137, 139, 140, 142–145, 147
\Count 157
\Cumulative 136
\cup 128
\CurrentOption 16

D

\DeclareOption 2, 4, 7–9, 11, 12, 14–16
\defaultfontfeatures 37
\ding 46–54
\displaystyle 191
\Distinct 137
\Domain 138
\draw 75

E

\Element 139
\Else 105, 112
\else 23, 42, 87, 104, 151, 165, 177
\end 79
\ENDFOR 161
\EndIf 159, 170
\Endtry 160, 161
\ensuremath . . . 176, 178
\envelope 49
\ExecuteOptions . . . 17
\Exists 162, 163
\exists 163

F

\False 106
\fi 27, 28, 55, 67, 89, 98, 108, 154, 167, 179
\filldraw 76, 77
\Floor 109
\footnotesize . . . 91, 92
\FORALL 194
\ForAll 166, 168
\forall 168
\Function 169

G

\geq 110
\GeqLex 110
\gets 176
\GlobalCardinality . 140
\green 82

H

\handpoint 39, 50
\handwrite 52
\hfill 26
\HigherOrder 141
\huge 71

I

\If 107, 112, 170
\if@astra@alg 5, 20
\if@astra@algiie 6, 21, 95, 103, 150, 164
\if@astra@gets . 13, 175
\if@astra@hyperref 10, 65
\if@astra@todo . . . 1, 85
\if@astra@xetex . . 3, 33
\Iff 111
\IfThen 170
\IfThenElse 112
\Implies 113
\In 116, 171, 172
\in 172
\Inequalities 142
\Integer 173
\Inter 114
\IntValuePrecedeChain 143
\Invariant 174
\IsAssigned . . . 176, 178

K

\Keyword 93, 94, 116, 152, 153, 157, 160, 174, 183, 184, 192, 193, 196

L

\langle 127
\lceil 102
\left 100, 102, 109, 119, 121, 122, 127, 141
\Leftrightarrow 111, 146
\leq 115
\LeqLex 115
\Let 116
\let 90
\Lex . . 110, 115, 117, 118
\lfloor 109
\Linear 144
\LtLex 118
\lvert 100

M

\marginpar 90
\mathcal 119
\mathtoolsset 60
\Maximise 181
\mid 124
\Minimise 182

N

\NbDistinct 145
\newfontinstance . . 38
\newif . 1, 3, 5, 6, 10, 13
\newKeyword . . 94, 96, 105–107, 120, 125, 126, 155, 156, 158, 159, 162, 166, 169, 171, 173, 180–182, 185–189, 195
\newtheorem 32
\node 78
\notick 41, 54

O

\Of 180
\Oh 119
\oldmarginpar . . . 90, 91

P

\PackageWarning . . . 16
\phone 47
\PostCond 183
\PreCond 184

\Procedure 185
\ProcessOptions . . . 18

R

\raggedleft 91
\raggedright 92
\rangle 127
\rceil 102
\red 82, 86, 141
\Reifies 146
\relax 18
\renewcommand 26, 91, 96, 152, 153
\RequirePackage 22, 24, 25, 29–32, 34, 35, 43, 45, 56–64, 66
\rfloor 109
\right 100, 102, 109, 119, 121, 122, 127, 141
\Rightarrow 113
\rvert 100

S

\scalebox 73
\scissors 46
\Select 120
\Sequence 121
\Set 122, 123
\SetComp 123
\SetDataSty 97
\SetDecl 186
\setdefaultlanguage 36
\SetKwData 96
\SetOf 187
\Size 188
\Solve 189
\STATE 192
\SuchThat 123, 124
\SUM 190, 191
\Sum 147
\sum 191

T

\text 117, 138, 148
\textbf 93, 97
\textsc 135
\Then 112, 125, 170
\tick 40, 53
\tikzstyle 71
\todo 85
\True 126
\Try 192
\Tryall 193, 194
\Tuple 127

U

\Union 128
\usetikzlibrary . . . 64

V

\Var 195
\Variables 148
\Variant 196
\victory 51