

Typesetting Deductions in L^AT_EX*

Michael Kohlhase
Computer Science, Jacobs University
<http://kwarc.info/kohlhase>

October 17, 2015

Abstract

Proof trees are usually presented in a table-like fashion in logic, the `ded` package provides an infrastructure for typesetting them in L^AT_EX.

Contents

1	The User Interface	2
1.1	Inference Rules	2
1.2	Derivation Trees	2
1.3	Display Environments	3
1.4	Tableaux	4
1.5	Miscellanea	4
1.6	Acknowledgements	4
2	The Implementation	5
2.1	Inference Rules and Derivations	5
2.2	Display Environments	8
2.3	Tableaux	9
2.4	Miscellanea	11

*Version v1.2 (last revised 2012/12/17)

The `ded` package provides an infrastructure for marking up complex proof trees and formal derivations. See Figure 1 for an example.

$$\frac{\frac{u : A \otimes B \vdash A \otimes B}{\frac{\frac{y : B \vdash B}{x : A, y : B, w : A \multimap B \multimap C \vdash C} \multimap E^{x,y}}{w : A \multimap B \multimap C, u : A \otimes B \vdash C} \multimap I^u}{\cdot \vdash (A \multimap B \multimap C) \multimap A \otimes B \multimap C} \multimap I^w$$

1.1 Inference Rules

\ian

\ianc

EdN:1

$$\frac{\text{ass}_1 \quad \dots \quad \text{ass}_n}{\text{concl}} \text{ name}$$

\ibn

\icn

\idn

The `\i*n` macro is useful in displaying single rules of inference, but it can do much more. I can be applied recursively to display derivation trees such as the one in Figure 2.

There are four variants of the basic `\ian` macro which differ by bounding box; we present this for `\ian*` here:

$$\backslash\mathrm{ian}\{a,b\}\{c\}\{n\} \quad \backslash\mathrm{ianc}\{a,b\}\{c\}\{n\} \quad \backslash\mathrm{ianl}\{a,b\}\{c\}\{n\} \quad \backslash\mathrm{ianr}\{a,b\}\{c\}\{n\}$$

¹EDNOTE: sounds like we could really automate this.

<pre> \begin{Displaynd} \icn{\ianr{\whyp{h}{\vdots}{a_1}{},a_2,a_3}{c}r} {\subtreec{A_1,A_2}{C}R} F D E \end{Displaynd} </pre>	$ \begin{array}{c} [h] \\ \vdots \\ \hline \frac{a_1 \quad a_2 \quad a_3}{c} r \quad \frac{A_1 \quad A_2}{C} R \quad F \\ \hline D \quad E \end{array} $
--	---

Example 2: A Derivation Tree built from nested inference rules

Of course the bounding box variants `\ibn*`, `\icn*`, `\idn*`, and `\ien*` also exist for $* \in \{c, l, r\}$.

The main purpose of the bounding box is to place subtrees in a derivation. In Figure 2 we use the `\whyp` macro to mark up a proof with a local hypothesis (commonly used in calculi of natural deduction). `\whyp[⟨label⟩]{⟨hyp⟩}{⟨pf⟩}{⟨conc⟩}{⟨rule⟩}` marks up a proof with hypothesis `⟨hyp⟩` with optional label `⟨label⟩` with body proof `⟨pf⟩` and conclusion `⟨conc⟩` justified by inference rule `⟨rule⟩`. Consider the two examples in Figure 3. In the first we have marked up the proposition implication introduction rule and in the second we have used that rule to prove the propositional K axiom (as we have two local hypotheses we made use of the possibility to give them a label to distinguish them).

<pre> \begin{displaynd} \newcommand{\imp}{\Rightarrow} \newcommand{\impI[1]{\Rightarrow\kern-.3em I^{#1}}} \whyp{A} {\ianc{\vdots}{B}{}} {A \imp B} {\impI{}} \end{displaynd} \begin{displaynd} \whyp[1]{A} {\whyp[2]{B}{A}{A \imp B}{\impI2}} {A \imp B \imp A} {\impI1} \end{displaynd} </pre>	$ \begin{array}{c} [A] \\ \vdots \\ \hline B \\ \hline A \Rightarrow B \Rightarrow I \end{array} $ $ \begin{array}{c} [A]^1 \\ [B]^2 \\ \hline A \\ \hline A \Rightarrow B \Rightarrow I^2 \\ \hline A \Rightarrow B \Rightarrow A \Rightarrow I^1 \end{array} $
---	---

Example 3: Derivations with local Hypotheses

For representing subtrees — i.e. proof trees we do not want to show — we have a macro `\subtree` that has 2 lines and a name for the subtree.

1.3 Display Environments

Even though inference rules can in principle be invoked any where in math mode, the `ded` package supplies a couple of specialized environments that initialize.

The `displaynd` and `Displaynd` environments allow for centered displayed derivation trees, much like the usual display style mathematics. The first leaves a bit more space between inference rules. The `cboxnd` is a variant that closes this in a box.

The `textnd` and

Finally, the environment is a variant of the L^AT_EX `figure` environment for derivation trees.

`fignd`

1.4 Tableaux

The `tableau` and `displaytableau` environments are used to start tableaux in in text and math modes. The `branches` environment is a variant of `array`; the `format` string is used to make the branches divided by a vertical line. Branches are separated by the `\nextbranch` macro, see Figure 4. The `\littrue` and `\litfalse` mark up true and false literals.

`tableau`
`displaytableau`
`branches`
`\nextbranch`
`\littrue`
`\litfalse`

```
\begin{displaytableau}
  \litfalse{(P\wedge Q)\Rightarrow(Q\wedge P)} \\\
  \littrue{P\wedge Q} \\\
  \litfalse{Q\wedge P} \\\
  \littrue{P} \\\
  \littrue{Q} \\\
  \begin{branches}{c|c}
    \litfalse{P} \\\bot
    \nextbranch
    \litfalse{Q} \\\bot
  \end{branches}
\end{displaytableau}
```

Example 4: A Tableau

1.5 Miscellanea

The `\rulename` can be used to construct names of inference rules for a calculus $\langle \textit{Calc} \rangle$ using `\rulename{\langle \textit{Calc} \rangle}{\langle \textit{name} \rangle}`.

`\rulename`

In some calculi that deal with computational linguistics we use the `\inputlf` macro for designing the input logical form.

`\inputlf`

1.6 Acknowledgements

An early version and the heart of the placement macro that is now `\inrulehelp` has been obtained from Frank Pfenning. The current `ded` package mainly adds periphery and documentation.

Florian Rabe has added functionality for hypothetical reasoning and Hella Hoffmann noticed an interface flaw that led to the introduction of `\i*nl` and `\i*nr`.

2 The Implementation

First we set up the lengths sizes.

```
1 (*package)
2 \def\@lineskipamount{4pt}
3 \def\@interlineskipamount{2pt}
```

\mud centers its argument in the current box.

```
4 \def\mud#1{\hfil $\displaystyle{#1}$\hfil}
```

\rig puts its argument to the right.

```
5 \def\rig#1{\hfil $\displaystyle{#1}$}
```

2.1 Inference Rules and Derivations

We first reserve some token registers and introduce a new switch, which we will use to create double lines if necessary.

```
6 \newbox\conc@box
7 \newbox\line@box
8 \newbox\name@box
9 \newbox\prem@box
10 \newbox\max@box
11 \newif\ifdouble\doublefalse
```

\inrulehelp The \inrulehelp macro is the main work horse for layouting the deduction trees. It constructs the layout for the upper part of the inference rule (including the horizontal bar and rule name) and saves it in the token register \conc@box. Then it constructs the conclusion and saves it in the box \line@box for later use. The

```
12 \newdimen\over@hang
13 \newdimen\tmp@dimen
14 \newdimen\max@wd
15 \newif\ifmax\maxfalse
16 \def\inrulehelp#1#2#3{%
17   \setbox\conc@box=\hbox{$\displaystyle{\mathstrut #2}$}%
18   \setbox\name@box=\hbox{$\; #3$}%
19   \setbox\line@box=\vbox{\vskip 2pt\halign{##\cr
20     \let\@tmpop=\relax
21     \mud{\@for\@I:={#1}\do{\@tmpop\@I\let\@tmpop=\quad}}\cr
22     \noalign{\vskip\the\lineskip}%
23     \noalign{\hrule height 0pt}%
24     \rig{\vbox to 0pt{\vss\hbox to 0pt{\copy\name@box \hss}\vss}}\cr
25     \noalign{\hrule}%
26     \ifdouble\noalign{\vskip\@interlineskipamount}\noalign{\hrule}\fi%
27     \noalign{\vskip\the\lineskip}%
28     \mud{\copy\conc@box}\cr}}%
29 \advance\max@wd by \wd\name@box
30 \tmp@dimen=\wd\line@box%
31 \advance\tmp@dimen by -\wd\conc@box%
32 \over@hang=.5\tmp@dimen}
```

\i*n

```
33 \def\ian#1#2#3{\lineskip\@lineskipamount%
34 \ifmax\inrulehelp{#1}{#2}{#3}\box\line@box\else%
35 \inrulehelp{#1}{#2}{#3}\hbox to \wd\conc@box{\hss\box\line@box\hss}\fi}
36 \def\ibn#1#2#3#4{\ian{#1}{#2}{#3}{#4}} %prem1, prem2, conc, name
37 \def\icn#1#2#3#4#5{\ian{#1}{#2}{#3}{#4}{#5}}
38 \def\idn#1#2#3#4#5#6{\ian{#1}{#2}{#3}{#4}{#5}{#6}}
```

```

39 \def\ien#1#2#3#4#5#6#7{\ian{#1},{#2},{#3},{#4},{#5}}{#6}{#7}}
40 \</package>
41 \*ltxml)
42 DefConstructor(' \ian{}{}{}', "<ltx:XApp>#3#1#2</ltx:XApp>");
43 DefConstructor(' \ibn{}{}{}{}', "<ltx:XApp>#4#1#2#3</ltx:XApp>");
44 DefConstructor(' \icn{}{}{}{}{}', "<ltx:XApp>#5#1#2#3#4</ltx:XApp>");
45 DefConstructor(' \idn{}{}{}{}{}{}', "<ltx:XApp>#6#1#2#3#4#5</ltx:XApp>");
46 DefConstructor(' \ien{}{}{}{}{}{}{}', "<ltx:XApp>#7#1#2#3#4#5#6</ltx:XApp>");
47 \</ltxml>

```

\i*nc

```

48 \<package>
49 \def\ianc#1#2#3{\lineskip\@lineskipamount%
50 \ifmax\inrulehelp{#1}{#2}{#3}\box\lin@box\else%
51 \inrulehelp{#1}{#2}{#3}\box\line@box\hskip\wd\name@box\fi}
52 \def\ibnc#1#2#3#4{\ianc{#1},{#2}}{#3}{#4}} %prem1, prem2, conc, name
53 \def\icnc#1#2#3#4#5{\ianc{#1},{#2},{#3}}{#4}{#5}}
54 \def\idnc#1#2#3#4#5#6{\ianc{#1},{#2},{#3},{#4}}{#5}{#6}}
55 \def\ienc#1#2#3#4#5#6#7{\ianc{#1},{#2},{#3},{#4},{#5}}{#6}{#7}}
56 \</package>
57 \*ltxml)
58 Let(' \ianc', '\ian');
59 Let(' \ibnc', '\ibn');
60 Let(' \icnc', '\icn');
61 Let(' \idnc', '\idn');
62 Let(' \ienc', '\ien');
63 \</ltxml>

```

\i*nl

```

64 \<package>
65 \def\ianl#1#2#3{\lineskip\@lineskipamount%
66 \ifmax\inrulehelp{#1}{#2}{#3}\box\line@box\else%
67 \inrulehelp{#1}{#2}{#3}\hskip\overhang\hbox to \wd\conc@box{\hss\box\line@box\hss}\fi}
68 \def\ibnl#1#2#3#4{\ianl{#1},{#2}}{#3}{#4}} %prem1, prem2, conc, name
69 \def\icnl#1#2#3#4#5{\ianl{#1},{#2},{#3}}{#4}{#5}}
70 \def\idnl#1#2#3#4#5#6{\ianl{#1},{#2},{#3},{#4}}{#5}{#6}}
71 \def\ienl#1#2#3#4#5#6#7{\ianl{#1},{#2},{#3},{#4},{#5}}{#6}{#7}}
72 \</package>
73 \*ltxml)
74 Let(' \ianl', '\ian');
75 Let(' \ibnl', '\ibn');
76 Let(' \icnl', '\icn');
77 Let(' \idnl', '\idn');
78 Let(' \ienl', '\ien');
79 \</ltxml>

```

\i*nr

```

80 \<package>
81 \def\ianr#1#2#3{\lineskip\@lineskipamount%
82 \ifmax\inrulehelp{#1}{#2}{#3}\box\line@box\else%
83 \inrulehelp{#1}{#2}{#3}\hbox to \wd\conc@box{\hss\box\line@box\hss}\hskip\overhang\hskip\wd\name@box\fi}
84 \def\ibnr#1#2#3#4{\ianr{#1},{#2}}{#3}{#4}} %prem1, prem2, conc, name
85 \def\icnr#1#2#3#4#5{\ianr{#1},{#2},{#3}}{#4}{#5}}
86 \def\idnr#1#2#3#4#5#6{\ianr{#1},{#2},{#3},{#4}}{#5}{#6}}
87 \def\ienr#1#2#3#4#5#6#7{\ianr{#1},{#2},{#3},{#4},{#5}}{#6}{#7}}
88 \</package>
89 \*ltxml)
90 Let(' \ianr', '\ian');

```

```

91 Let('ibnr','ibn');
92 Let('icnr','icn');
93 Let('idnr','idn');
94 Let('ienr','ien');
95 
```

EdN:2 \i*nm 2

```

96 (*package)
97 \def\ianm#1#2#3{\lineskip\@lineskipamount\maxtrue\inrulehelp{#1}{#2}{#3}\box\line@box}
98 
```

\subtree

```

99 (*package)
100 \def\subtree#1#2#3{\doubletrue\ian{#1}{#2}{#3}}
101 \def\subtreec#1#2#3{\doubletrue\ianc{#1}{#2}{#3}}
102 
```

\small/normalnd

```

107 (*package)
108 \def\smallnd{\def\Rulespacing{\renewcommand{\arraystretch}{3}\arraycolsep 0em}}
109 \def\normalnd{\def\Rulespacing{\renewcommand{\arraystretch}{4}\arraycolsep 0em}}
110 \normalnd
111 \def\normalspacing{\renewcommand{\arraystretch}{1}}
112 
```

EdN:3 3

\dedover

```

118 (*package)
119 \def\dedover#1#2{\hbox{\vbox{\displaystyle{\mathstrut
120 #1}\displaystyle{\mathstrut #2}}}}
121 
```

\matop

```

125 (*package)
126 \def\matop#1{\def\arraystretch{1}\begin{array}{c}#1\end{array}}
127 
```

\ded@hyp \ded@hyp[*label*]{*hyp*}marks up a hypothesis *hyp* with optional label *label*.

```

131 (*package)
132 \newcommand\ded@hyp[2][\def\@test{#1}\left[#2\right]\ifx\@test\empty\else^{#1}\fi
```

²EdNOTE: MK: this needs to be documented

³EdNOTE: the following need to be consolidated

`\whyp*` `\whyp*[\langle label \rangle]{\langle hyp \rangle}{\langle pf \rangle}{\langle conc \rangle}{\langle rule \rangle}` marks up a proof with hypothesis $\langle hyp \rangle$ with optional label $\langle label \rangle$ with body proof $\langle pf \rangle$ and conclusion $\langle conc \rangle$ justified by inference rule $\langle rule \rangle$.

```
133 \newcommand\whyp[5] [] {\ian{\ded@atop[0]{\ded@hyp[#1]{#2}}{#3}}{#4}{#5}}
134 \newcommand\whypc[5] [] {\ianc{\ded@atop[0]{\ded@hyp[#1]{#2}}{#3}}{#4}{#5}}
135 \newcommand\whyp1[5] [] {\ianl{\ded@atop[0]{\ded@hyp[#1]{#2}}{#3}}{#4}{#5}}
136 \newcommand\whyp1r[5] [] {\ianr{\ded@atop[0]{\ded@hyp[#1]{#2}}{#3}}{#4}{#5}}
```

`\hypjuda` `\hypjuda{h}{c}` for a hypothetical judgment with hypothesis h and conclusion c , `\hypjudb` for two hypotheses etc.⁴

EdN:4

```
137 \newcommand{\ded@atop}[3] [] {\genfrac{}{}{0pt}{#1}{#2}{#3}}
138 \newcommand{\hypjuda}[3] [] %
139 {\def\@test{#1}\ded@atop[0]{\ded@atop[0]{\ded@hyp[#1]{#2}}{\vdots}}{#3}}
140 \newcommand{\hypja}[3] [] {\def\@test{#1}\ded@atop[0]{\ded@hyp[#1]{#2}}{#3}}
141 \newcommand{\hypjudb}[4] [] {\ded@atop[0]{\ded@hyp[#2]}{\hypjuda[#1]{#3}{#4}}}
142 \newcommand{\hypjudc}[5] [] {\ded@atop[0]{\ded@hyp[#2]}{\hypjudb[#1]{#3}{#4}{#5}}}
143 \end{package}
144 \end{xml}
145 \DefConstructor('hypjuda[]{}{}', '<ltx:XMAp>#3#2</ltx:XMAp>');
146 \DefConstructor('hypja[]{}{}', '<ltx:XMAp>#3#2</ltx:XMAp>');
147 \DefConstructor('hypjudb[]{}{}{}', '<ltx:XMAp>#4#3#2</ltx:XMAp>');
148 \DefConstructor('hypjudc[]{}{}{}{}', '<ltx:XMAp>#5#4#3#2</ltx:XMAp>');
149 \end{xml}
```

2.2 Display Environments

We first set some lengths

```
150 (*package)
151 \def\Displaynd@pre@space{0em}
152 \def\Displaynd@post@space{-1em}
153 \def\cbox@pre@space{-.5em}
154 \def\cbox@post@space{-2.5em}
155 \def\cbox@left@space{.3em}
156 \def\cbox@right@space{.3em}
157 \end{package}
```

`displaynd`

```
158 (*package)
159 \newenvironment{displaynd}%
160 {\begin{displaymath}\Rulespacing\begin{array}{c}}%
161 {\end{array}\end{displaymath}\aftergroup\ignorespaces}
162 \end{package}
163 \end{xml}
164 \DefEnvironment('displaynd', '<ltx:Math><ltx:XMAp>#body</ltx:XMAp></ltx:Math>', mode=>'display_math');
165 \end{xml}
```

`Displaynd`

```
166 (*package)
167 \newenvironment{Displaynd}[1]%
168 {\vspace*{\Displaynd@pre@space}\begin{displaymath}\Rulespacing\begin{array}{#1}}%
169 {\end{array}\end{displaymath}\aftergroup\ignorespaces\vspace*{\Displaynd@post@space}}
170 \end{package}
171 \end{xml}
172 \DefEnvironment('Displaynd', '<ltx:Math><ltx:XMAp>#body</ltx:XMAp></ltx:Math>', mode=>'display_math');
173 \end{xml}
```

⁴EDNOTE: the latexml implementation is completely wrong, need a binder there.

textnd

```
174 (*package)
175 \newenvironment{textnd}%
176 {$\displaystyle\Rulespacing\begin{array}{c}}%
177 {\end{array}$}
178 \end{package}
179 \end{ltxml}
180 DefEnvironment('{textnd}', '<ltx:Math><ltx:XMath>#body</ltx:XMath></ltx:Math>', mode=>'inline_math');
181 \end{ltxml}
```

\ndsepline

```
182 (*package)
183 \def\ndsepline{\hline\[-7ex]}
184 \end{package}
185 \end{ltxml}
186 \end{ltxml}
```

cboxnd

```
187 (*package)
188 \newenvironment{cboxnd}%
189 {\vspace*{\cbox@pre@space}
190 \begin{displaymath}\Rulespacing
191 \begin{array}{|@{\hspace{\cbox@left@space}}c@{\hspace{\cbox@right@space}}|\}\hline}%
192 {\hline\end{array}\end{displaymath}
193 \aftergroup\ignorespaces
194 \vspace*{\cbox@post@space}}
195 \end{package}
196 \end{ltxml}
197 DefEnvironment('{cboxnd}', '<ltx:Math><ltx:XMath>#body</ltx:XMath></ltx:Math>', mode=>'inline_math');
198 \end{ltxml}
```

tboxnd

```
199 (*package)
200 \newenvironment{tboxnd}%
201 {$\displaystyle\Rulespacing
202 \begin{array}{|@{\hspace{\cbox@left@space}}c@{\hspace{\cbox@right@space}}|\}\hline}%
203 {\hline\end{array}$}
204 \end{package}
205 \end{ltxml}
206 DefEnvironment('{tboxnd}', '<ltx:Math><ltx:XMath>#body</ltx:XMath></ltx:Math>', mode=>'inline_math');
207 \end{ltxml}
```

fignd

```
208 (*package)
209 \newenvironment{fignd}[2]%
210 {\begin{figure}[htb]\def\fignd@label{fig:#1}\def\fignd@capt{{#2}}\begin{cboxnd}}%
211 {\end{cboxnd}\caption{\fignd@capt}\label{\fignd@label}\end{figure}}
212 \def\endfignd{\end{cboxnd}\caption{\fignd@capt}\label{\fignd@label}\end{figure}}
213 \end{package}
214 \end{ltxml}
215 DefEnvironment('{fignd}', '<ltx:Math><ltx:XMath>#body</ltx:XMath></ltx:Math>', mode=>'inline_math');
216 \end{ltxml}
```

2.3 Tableaux

tableau Tableaux are modeled as arrays in \LaTeX , and as nested proof terms in \LaTeXML . For the latter we need to disable the newline macro.

```

217 <*package>
218 \newenvironment{tableau}%
219 {\arraycolsep .2em\def\arraystretch{.9}\begin{array}{c}}%
220 {\end{array}}
221 </package>
222 <*ltxml>
223 DefEnvironment(' {tableau}',
224               '<ltx:XMAp><ltx:XTok meaning="tableau" omcd="tableaux"/>#body</ltx:XMAp>',
225               beforeDigest=>sub { DefMacro("\\\\", ''); } );
226 </ltxml>

```

displaytableau*

```

227 <ltxml>RawTeX('
228 <*package | ltxml>
229 \newenvironment{displaytableau*}%
230 {\begin{displaymath}\begin{tableau}}%
231 {\end{tableau}\end{displaymath}\aftergroup\ignorespaces}
232 </package | ltxml>
233 <ltxml>');

```

displaytableau

```

234 <*package>
235 \newenvironment{displaytableau}[1]% label
236 {\begin{equation}\label{tab:#1}\begin{tableau}}%
237 {\end{tableau}\end{equation}\aftergroup\ignorespaces}
238 </package>
239 <*ltxml>
240 DefEnvironment(' {displaytableau}{}', '<ltx:Math><ltx:XMATH>#body</ltx:XMATH></ltx:Math>',
241               mode=>'display_math');
242 </ltxml>

```

branches

```

243 <*package>
244 \newenvironment{branches}[1]% formt
245 {\begin{array}[t]{#1}\begin{array}[t]{c}}%
246 {\end{array}\end{array}}
247 </package>
248 <*ltxml>
249 DefEnvironment(' {branches}{}',
250               '<ltx:XMAp><ltx:XTok meaning="branches" omcd="tableaux"/>'
251               . '<ltx:XMAp><ltx:XTok meaning="tableau" omcd="tableaux"/>'
252               . '#body'
253               . '</ltx:XMAp>'
254               . '</ltx:XMAp>');
255 </ltxml>

```

\nextbranch

```

256 <package>\def\nextbranch{\end{array}&\begin{array}[t]{c}}
257 <*ltxml>
258 DefConstructor('\nextbranch', '</ltx:XMAp><ltx:XMAp><ltx:XTok meaning="tableau" omcd="tableaux"/>');
259 </ltxml>

260 <*package>
261 \def\litfalse#1{#1^{\mathsf{f}}}
262 \def\littrue#1{#1^{\mathsf{t}}}
263 </package>

```

2.4 Miscellanea

EdN⁵
264 (*package)
265 \def\rulename#1#2{\mbox{\sf{#1:#2}}}
266 \end{package}
267 (*ltxml)
268 DefConstructor('\rulename{}{}', '<ltx:XTok omcd="FIXME" meaning="#1-#2"/>');
269 \end{ltxml}

EdN⁶
270 (*package)
271 \def\inputlf#1{\fbox{\ensuremath{#1}}}
272 \end{package}
273 (*ltxml)
274 DefConstructor('\inputlf{}', '#1');
275 \end{ltxml}

Finally, we need to terminate the file with a success mark for perl.

276 \end{ltxml}1;

⁵EDNOTE: fix the current theory

⁶EDNOTE: fix me!