

Mixfix injections in free types

[/Reference manual/Non-standard extensions](#)

Within free type paragraphs, injections are traditionally written using $\langle\langle$ and $\rangle\rangle$ brackets. The name of an injection can be a functional operator. CADiZ permits such injections to be written using the mixfix syntax of an application. For example (ignoring overloading of existing Z symbols)...

```
function 103 ( _ N)
function 101 leftassoc ( _ + _ )
function 101 leftassoc ( _ - _ )
function 102 leftassoc ( _ * _ )
function 102 leftassoc ( _ / _ )
```

$$\begin{aligned} term ::= & \mathbb{A} \ N \\ & | \mathbb{A} + \mathbb{A} \\ & | \mathbb{A} - \mathbb{A} \\ & | \mathbb{A} * \mathbb{A} \\ & | \mathbb{A} / \mathbb{A} \end{aligned}$$

The effect of this on the concrete syntax is to redefine *Branch* as follows, where the second production covers both the new syntax of injections and the old syntax of elements.

$$\begin{aligned} Branch &= DeclName, \langle\langle, Expression, \rangle\rangle \\ &| Expression \\ &; \end{aligned}$$



IT 4-Jan-2002

Page 1 of 2



Go Back

Full Screen

Close

Quit