

one-point

[/Reference manual/Z-related commands/In situ replacement commands](#)

The *one-point* command eliminates those local variables for which there is an outermost conjunct equating it to an expression: for each such variable, all references to it are replaced by the equal expression, and the variable's declaration is removed. In the following example, *one-point* eliminates the variable x .

$$\exists x, y : \mathbb{N} \mid y \leq x \wedge x = 4 \bullet y > 2 \implies \exists y : \mathbb{N} \mid 4 \in \mathbb{N} \wedge y \leq 4 \bullet y > 2$$

Note that the property implicit in the declaration of x , i.e. $x \in \mathbb{N}$, gives rise on elimination of x to the conjunct $4 \in \mathbb{N}$ in the result.

The *one-point* command can be used with quantified predicates, set comprehension, definite description and let expressions. In each case, the outermost conjuncts of the predicate in the \mid part of the schema text are candidates for exploitation by *one-point*. Also, the equalities implicit in $==$ declarations are treated as candidates for exploitation by *one-point*. The outermost conjuncts of the predicate in the \bullet part of existentially quantified predicates, and the outermost conjuncts of the implicand of the predicate in the \bullet part of universally quantified predicates where that predicate is an implication, are also candidates for exploitation by *one-point*.

Schema declarations can be eliminated if there is an equality between theta expressions, one of whose operand schema expression is identical to the declaration's schema expression. For example,

$$\exists S; y : \mathbb{N} \mid \theta S = \theta S' \bullet x > y \implies \exists y : \mathbb{N} \mid \theta S' \in S \bullet x' > y$$

Again, the property implicit in the declaration is retained.

Having done as many eliminations of variables as are possible, *one-point* restores those properties that were originally implicit in the declarations of remaining variables.

The *one-point* command cannot eliminate variables that have multiple (merged) declarations. Equalities whose variable on one side is also referred to by the expression on the other side cannot be, and are not, used. Variables for which there are implicit uses concealed in theta expressions and schema predicates cannot be eliminated, except by equalities between theta expressions where the effect can be expressed by renaming. The *one-point* command cannot be instructed to eliminate only particular variables, nor is it possible to articulate which equality is used in the case when there are several equalities for the same variable. The *Leibniz* command may help in such circumstances.

1. Tactic example

“one-point” $p_1 \ p_2$

This example applies the *one-point* command to predicates p_1 and p_2 .

IT 21-Dec-1998