

rewrite by rule

[/Reference manual/Z-related commands/Proof rule commands](#)

Page 1 of 2



Go Back

Full Screen

Close

Quit

The *rewrite by rule* command applies a rewrite rule within a selection until it is no longer applicable. The rewrite rule must be a named conjecture. (The name is needed for *record script*.) The conjecture can have generic parameters, but no hypotheses. It must have a single consequent. That consequent must be a rewrite rule as defined for *rewrite by antecedent*.

The selection can be an expression, a predicate, a schema text, or a whole goal. Rewriting is done as for *rewrite by antecedent*, except that all sub-formulae within the selection are candidates for rewriting rather than just the whole selection, and further rewrites are done on the result until the rewrite rule no longer matches any sub-formula. If the | part of the schema text to which generated side-conditions are conjoined is within the selection, then those side-conditions are themselves candidates for rewriting.

When applied interactively, the rewrite rule is selected in the specification window. Any selection in the left window is also taken into account, so if you want to rewrite a selection in the specification window, any selection in the left window must be cleared first.

A conjecture cannot be used as a rewrite rule in its own proof.

1. Tactic examples

Page 1 of 2



Go Back

Full Screen

Close

Quit

“rewrite by rule” “rule_name” 0 e

This example applies the *rewrite by rule* command using the rewrite rule named “rule_name” within the expression *e* in the goal. (The 0 is needed to correspond to the selection of a rule, but is not in itself sufficient to identify which rule.)

IT 30-Mar-2000