"Less than" in terms of "at most"

If, a in the case of comparing reals, < is a total order, < is most simply defined by

(0) \[ x < y \equiv \neg (y \leq x) \]

The "transitivity" rule

(1) \[ a \leq b \land b < c \Rightarrow a < c \]

is then a direct consequence of the transitivity of \( \leq \):

\[
\begin{align*}
a \leq b \land b < c & \Rightarrow a < c \\
= & \quad \{ (0), \text{ twice} \} \\
a \leq b \land \neg (c \leq b) & \Rightarrow \neg (c \leq a) \\
= & \quad \{ \text{shunting, twice} \} \\
c \leq a \land a \leq b & \Rightarrow c \leq b \\
= & \quad \{ \leq \text{ transitive} \} \\
& \text{true}
\end{align*}
\]

If \( \leq \) is not a total order, this does not work, and I used to use the definition

(2) \[ x < y \equiv x \leq y \land x \neq y \]

The proof of (1) is then as follows:

\[
\begin{align*}
a \leq b \land b < c & \Rightarrow a < c \\
= & \quad \{(2), \text{ twice}\} \\
a \leq b \land b \leq c \land b \neq c & \Rightarrow a \leq c \land a \neq c \\
= & \quad \{ \leq \text{ is transitive} \}
\end{align*}
\]
\[ a \leq b \land b \leq c \land b \neq c \Rightarrow a \neq c \]

= \{ \text{shunting, twice} \}
\[ a \leq b \land b \leq c \land a = c \Rightarrow b = c \]

= \{ \text{Leibniz} \}
\[ c \leq b \land b \leq c \land a = c \Rightarrow b = c \]

= \{ \leq \text{ reflexive} \}
true

In [GS94] - "A logical approach to discrete math" by Gries & Schneider - I found the much nicer definition:

(3) \[ x \lessdot y \equiv x \lessdot y \land \neg(y \lessdot x) \]

The proof of (1) is now very clean:

\[ a \leq b \land b \leq c \Rightarrow a \leq c \]

= \{ (3), twice \}
\[ a \leq b \land b \leq c \land \neg(c \leq b) \Rightarrow a \leq c \land \neg(c \leq a) \]

= \{ \text{pred. calc.} \}
\[ (a \leq b \land b \leq c \Rightarrow a \leq c) \land (c \leq a \land a \leq b \Rightarrow c \leq b) \]

= \{ \leq \text{ transitive, twice} \}
true.

The moral of the story is that the conjunct \( x \neq y \) in (2) is awkward: it forces in general, in one way or another, an appeal to Leibniz, and in this special case also an appeal to the reflexivity of \( \leq \). Definition (3) is much nicer because it
uses & only.

In retrospect I am amazed that I did not come up with (3) myself, for it is a straightforward strengthening of (0), with which I grew up.

Nuenen, 11 August 1994

prof. dr. Edsger W. Dijkstra
Department of Computer Sciences
The University of Texas at Austin
Austin, TX 78712-1188
USA