The Order of Mixed Quantifiers

For those who are having trouble understanding the "quantifier switch" fallacy, the following discussion should help.

When quantifiers in the same sentence are of the same quantity (all universal or all existential), the order in which they occur does not matter. But when they are mixed, the order in which they occur becomes crucial. Consider these examples:

$\forall x \forall y \text{ Likes}(x, y)$	\Leftrightarrow	∀y ∀x Likes(x, y)
∃x ∃y Likes(x, y)	\Leftrightarrow	∃y ∃x Likes(x, y)

These are clearly equivalent pairs. The first pair contains two different ways of saying *everyone likes everyone*. The second contains two different ways of saying *someone likes someone*.

Now consider this mixed quantifier case:

$$\forall x \exists y \text{ Likes}(x, y) \Leftrightarrow \exists y \forall x \text{ Likes}(x, y)$$

Clearly these are not equivalent sentences. The one on the left says (very plausibly) that everyone likes someone (or other), but allows for the possibility that different people have different likes—I like Edgar Martinez, you like Ken Griffey, Jr., Madonna likes herself, etc. The one on the right, however, says something much stronger—it says that there is at least one person so well liked that *everyone* likes him or her. (It's very unlikely that there is such a person, and so very unlikely that the sentence on the right is true.)

Notice that the stronger sentence (on the right) logically implies the weaker one (on the left). In general, an $\exists \forall$ sentence logically implies its $\forall \exists$ counterpart, but not conversely.

For a more dramatic contrast, consider this pair of sentences:

Again, these are not equivalent. The one on the left is a logical truth; it says *everything is identical to something*. The one on the right says *there is something such that everything is identical to that thing*, and this comes very close to being logically false. (It is not logically false, because there are at least some worlds in which it is true. Can you think of one?)