The Reflexivity Rule is extended automatically. If $\sigma\,\langle\lambda x.\Phi(x)\rangle(\imath y.\Psi(y))$ occurs on a tableau branch, we can add $\sigma\, p = [\imath y.\Psi(y)]^{\sigma}$ where p is a new parameter, using the Derived Existence Rule, then add $\sigma\,(p = p)$ using our earlier version of reflexivity, and finally we can add $\sigma\,[(\imath y.\Psi(y)]^{\sigma} = [\imath y.\Psi(y)]^{\sigma}$ using substitutivity of equality.

We are finished with cases that simply extend rules we have seen before. Next we have important new rules that make use of the special structural features of definite descriptions. Note that one of the rules involves three-way branching.

Definition 20.7.6 (Definite Description Rules, Positive, *CN*)

$$\frac{\sigma\,\langle\lambda x.\Phi(x)\rangle(\imath y.\Psi(y))}{\sigma\,\Psi([\imath y.\Psi(y)]^{\sigma})} \qquad \frac{\sigma\,\langle\lambda x.\Phi(x)\rangle(\imath y.\Psi(y))}{\sigma\,\neg\Psi(p) \mid \sigma\,\langle\lambda x.x = p\rangle(\imath y.\Psi(y))}$$
$$\textit{for any parameter } p$$

Definition 20.7.7 (Definite Description Rule, Negative, *CN*)

$$\frac{\sigma\,\neg\langle\lambda x.\Phi(x)\rangle(\imath y.\Psi(y))}{\sigma\,\neg\Psi(p) \;\Bigg|\; \begin{array}{l}\sigma\,\Psi(q)\\ \sigma\,\neg(p = q)\end{array} \;\Bigg|\; \sigma\,\langle\lambda x.x = p\rangle(\imath y.\Psi(y))}$$

For any parameter p, and new parameter q.

This completes the presentation of our tableau system. Several examples of it in use are in Section 20.8, soundness is discussed in Section 20.9, and completeness in Section 20.10. But before continuing, we discuss some of the intuitions behind the Positive and the Negative Definite Description Rules.

For the second of the Positive rules, if $\sigma\,\langle\lambda x.\Phi(x)\rangle(\imath y.\Psi(y))$ occurs on a tableau branch, informally we think of $\langle\lambda x.\Phi(x)\rangle(\imath y.\Psi(y))$ as being true at possible world σ. Then in particular, $\imath y.\Psi(y)$ must designate at σ, so if something makes $\Psi(y)$ true at σ, that something must be what $\imath y.\Psi(y)$ designates at σ. That is, if p is a parameter and $\Psi(p)$ holds at world σ then $\langle\lambda x.x = p\rangle(\imath y.\Psi(y))$ must hold. In effect, this implication is what the two parts of the consequence of the rule say when taken together. (Think of an implication as the disjunction of its negated antecedent with its consequent, and tableau branching as disjunction.)

The Negative rule is more complex since it has a three way split instead of two way. If $\sigma\,\neg\langle\lambda x.\Phi(x)\rangle(\imath y.\Psi(y))$ occurs on a branch, informally $\langle\lambda x.\Phi(x)\rangle(\imath y.\Psi(y))$ is false at possible world σ. It could be false because $\imath y.\Psi(y)$ does not designate at σ, or because it does designate but what is designated does not make $\Phi(x)$ true. Let p be any parameter. If p is not something that $\imath y.\Psi(y)$ designates at σ, then either this is because p does not make $\Psi(y)$ true (represented by the left hand branch), or because p and something else both make $\Psi(y)$ true (represented by the middle branch). If, somehow, we have eliminated both of these, then $\imath y.\Psi(y)$ must designate at σ, and p must be what it designates (this is represented by the right branch). Note that on the right branch, $\imath y.\Psi(y)$ is positively generated, so using the second Predicate Abstraction Rule from Definition 20.7.3 we can get that it designates but does not make $\Phi(x)$ true at σ.