15-399 Supplementary Notes: Definitional Equality

Robert Harper

February 17, 2004

1 Definitional Equality

The rules for simplifying terms derived from composing an eliminatory form with an introductory form give rise to an equivalence relation, called *definitional equality*, on terms, which is written $t \equiv u$. This relation is the least equivalence relation containing the reduction relation $t \leadsto u$. (Recall that reduction means "simplify anywhere" in a term.) More precisely, it is defined by the following rules:

$$\frac{t \leadsto u}{t \equiv u} \qquad \frac{}{t \equiv t}$$

$$\frac{u \equiv t}{t \equiv u} \qquad \frac{t \equiv u \quad u \equiv v}{t \equiv v}$$

We sometimes write $\Gamma \vdash t \equiv u \in \tau$ to mean $\Gamma \vdash t \in \tau$ and $\Gamma \vdash u \in \tau$ and $t \equiv u$.

2 Predicates and Truth

A *predicate* over a type is a property of the elements of that type. If p is a predicate over τ, and $t \in \tau$, then $p(t)$ is a proposition expressing that the property p holds of the element t. A *binary relation* is a predicate over a product type $\tau_1 \times \tau_2$; a *n-ary relation* is a predicate over $\tau_1 \times \cdots \times \tau_n$, where $n \geq 2$. For example, "even-ness" is a predicate over the type **nat**, and "less than" is a predicate over the type **nat** $\times$ **nat**. That is, "less than" is a binary relation over the type **nat**.

These are the rules for the formation and definitional equality of predicates:

$$\frac{\Gamma \vdash t \in \tau}{\Gamma \vdash p(t)\ \mathit{prop}} \qquad \frac{\Gamma \vdash t \equiv u \in \tau}{\Gamma \vdash p(t) \equiv p(u)\ \mathit{prop}}$$

Definitional equality extends to arbitrary propositions by these rules:

$$\frac{}{\Gamma \vdash \top \equiv \top \; \mathit{prop}} \qquad \frac{}{\Gamma \vdash \bot \equiv \bot \; \mathit{prop}}$$

$$\frac{\Gamma \vdash P \equiv P' \quad \Gamma \vdash Q \equiv Q'}{\Gamma \vdash P \wedge Q \equiv P' \wedge Q'} \qquad \frac{\Gamma \vdash P \equiv P' \quad \Gamma \vdash Q \equiv Q'}{\Gamma \vdash P \vee Q \equiv P' \vee Q'}$$

$$\frac{\Gamma \vdash P \equiv P' \quad \Gamma \vdash Q \equiv Q'}{\Gamma \vdash P \supset Q \equiv P' \supset Q'}$$

Finally, if a proposition P is true and definitionally equal to Q, then Q is also true:

$$\frac{\Gamma \vdash P \; \mathit{true} \quad \Gamma \vdash P \equiv Q \; \mathit{prop}}{\Gamma \vdash Q \; \mathit{true}}$$

3 Proofs and Terms

We may reduce logic to type theory by introducing the following definitional equalities that associates with each a proposition the type of its proofs:

$$\begin{array}{rcl} \mathbf{pfs}(\top) & \equiv & \mathbf{1} \\ \mathbf{pfs}(\bot) & \equiv & \mathbf{0} \\ \mathbf{pfs}(P \wedge Q) & \equiv & \mathbf{pfs}(P) \times \mathbf{pfs}(Q) \\ \mathbf{pfs}(P \vee Q) & \equiv & \mathbf{pfs}(P) + \mathbf{pfs}(Q) \\ \mathbf{pfs}(P \supset Q) & \equiv & \mathbf{pfs}(P) \rightarrow \mathbf{pfs}(Q) \end{array}$$

The proof-term judgment $M : P$ may be regarded as an abbreviation for $M \in \mathbf{pfs}(P)$. We have the following rules governing these types:

$$\frac{\Gamma \vdash P \; \mathit{prop}}{\Gamma \vdash \mathbf{pfs}(P) \; \mathit{type}} \qquad \frac{\Gamma \vdash P \equiv Q \; \mathit{prop}}{\Gamma \vdash \mathbf{pfs}(P) \equiv \mathbf{pfs}(Q)}$$

4 Families of Types

Since propositions can involve terms (through the basic predicates), we now have the possibility of types involving terms. For example, $\mathbf{pfs}(x =_N y)$ is a type, if $x, y \in \mathbf{nat}$. This is an example of a *family of types indexed by a type* (in this case, the type $\mathbf{nat} \times \mathbf{nat}$).

In general a *family of types* $\tau(x)$ *indexed by* $x \in \sigma$ determines a type $\tau(t)$ for each element $t \in \sigma$. Moreover, definitionally equivalent indices determine definitionally equivalent types:

$$\frac{\Gamma \vdash t \in \sigma}{\Gamma \vdash \tau(t) \; \mathit{type}} \qquad \frac{\Gamma \vdash t \equiv u}{\Gamma \vdash \tau(t) \equiv \tau(u) \; \mathit{type}}$$