

# TinyLF: A Compact Definition of a Logical Framework

Jason Reed

July 1, 2007

## 1 Introduction

By pushing some modern ideas concerning LF to their logical conclusion, one can obtain a very concise definition of a logical framework, which has a comparably simple metatheory. This logical framework generalizes LF slightly by including a limited form of polymorphism. The main point, however, is not this extension per se, but that we can support the *same* feature set as LF in a much more compact way.

The central ideas are naturally the recently ubiquitous notion of *restricting attention to only canonical forms* of  $\lambda$ -expressions, and the consequent need to define *hereditary substitution*, pioneered by Watkins et al in CLF.

Another main contributing idea is that of spine form, which considers not just single function applications but applications of heads to entire vectors of arguments at once. Spine form still describes function abstractions in terms of one-at-a-time  $\lambda$ -binding of variables. The motivation is to find a convenient syntax in which variable binding also takes place in a ‘vectorized’ fashion. This is obviously related to the idea of pattern-matching in functional languages. However, in the present case, given the usual normal/atomic distinction that serves to enforce  $\beta$ -normal,  $\eta$ -long terms, we are able to demand that functions provide the *entire* pattern of their arguments, coalescing all potential curried arguments into one sequence.

But the other salient recent piece of work where sequences of terms play an important role is the Contextual Modal Type Theory designed to give a logical account of operations such as raising and lowering of metavariables in unification. In it, modal variables appear under *substitutions* that fill in the variable’s arguments all at once. We take essentially the same approach while dropping the *closedness* requirement on substitutions for modal variables, making our variables still ‘contextual’ but not modal.

## 2 Syntax

The complete syntax of the language is as follows:

Base Classifiers	$v$	$::=$	$R \mid \text{type}$
Classifiers	$V$	$::=$	$\Pi\Psi.v$
Atomic Terms	$R$	$::=$	$x[\sigma]$
Terms	$M$	$::=$	$\lambda\hat{\Psi}.R$
Substitutions	$\sigma$	$::=$	$\text{id} \mid M.\sigma$
Contexts	$\Gamma$	$::=$	$\cdot \mid \Gamma, x : V$

The syntax  $\hat{\Psi}$  stands for just the sequence of variables found in  $\Psi$ , without their associated types. All contexts are to be understood as dependent, with  $V$ s on the right being in the scope of variables on their left. We will use  $=$  to mean syntactic equality up to  $\alpha$ -conversion.

The terms  $M$  of LF correspond to terms here; their consecutive  $\lambda$ s have simply been merged into one context. For example, the LF term

$$\lambda x.\lambda y.\lambda z.z$$

corresponds to

$$\lambda(x, y, z).z[\text{id}]$$

Variable occurrences are always *under substitution*, the substitution being the analogue to the spine attached to a root in spine form. Indeed, all substitutions are vectors of terms. To carry out a substitution, we will write  $\{\sigma/\Psi\}$ , substituting the items found in  $\sigma$  for the variables in  $\Psi$ . We call these term vectors substitutions rather than spines because they associate the opposite way.

Both the types and kinds of LF become *classifiers* in the present system. A base classifier is either a base type  $x[\sigma]$ , which corresponds to a base type  $a \cdot S$  in spine-form  $LF$ , or else it is the base kind type. This merger is reminiscent of (but not quite identical to) the uniformity typical of Pure Type Systems. The difference is that while we allow type variables  $x : \text{type}$  to appear in contexts, the only things that can be substituted for  $x$  are *base* types.

Because variables in the context can have classifiers such as  $\Pi\Psi.\text{type}$  that are essentially kinds, we have no need for a separate notion of signature: the context by itself can contain a mixture of declarations of type families and constants.

## 3 Typing

The judgments are

$\Gamma \vdash V \Leftarrow \text{ok}$	$V$ is a well-formed classifier
$\Gamma \vdash N \Leftarrow V$	$N$ checks at classifier $V$
$\Gamma \vdash R \Rightarrow v$	$R$ synthesizes classifier $v$
$\Gamma \vdash \sigma \Leftarrow \Psi$	$\sigma$ is a valid substitution for $\Psi$
$\Gamma \vdash \Psi \text{ ctx}$	$\Psi$ is a well-formed context

Well-formedness of classifiers is defined by two rules, one that allows formation of types and another that forms kinds:

$$\frac{\Gamma \vdash \Psi \text{ ctx} \quad \Gamma, \Psi \vdash R \Rightarrow \text{type}}{\Gamma \vdash \Pi\Psi.R \Leftarrow \text{ok}} \quad \frac{\Gamma \vdash \Psi \text{ ctx}}{\Gamma \vdash \Pi\Psi.\text{type} \Leftarrow \text{ok}}$$

The remaining judgments are defined by the following rules:

$$\frac{\Gamma, \Psi \vdash R \Rightarrow v' \quad v = v'}{\Gamma \vdash \lambda\hat{\Psi}.R \Leftarrow \Pi\Psi.v} \quad \frac{x : \Pi\Psi.v \in \Gamma \quad \Gamma \vdash \sigma : \Psi}{\Gamma \vdash x[\sigma] \Rightarrow v\{\sigma/\Psi\}}$$

$$\frac{}{\Gamma \vdash \text{id} \Leftarrow \cdot} \quad \frac{\Gamma \vdash M \Leftarrow V\{\sigma/\Psi\} \quad \Gamma \vdash \sigma \Leftarrow \Psi}{\Gamma \vdash M.\sigma \Leftarrow (\Psi, x : V)}$$

$$\frac{}{\Gamma \vdash \cdot \text{ ctx}} \quad \frac{\Gamma \vdash \Psi \text{ ctx} \quad \Gamma, \Psi \vdash V \Leftarrow \text{ok}}{\Gamma \vdash \Psi, x : V \text{ ctx}}$$

Let  $X$  stand for any syntactic object  $v, v, M, R, \sigma, \Gamma$ . All that remains to define is substitution  $X\{\sigma/\Psi\}$ . It is defined recursively according to a lexicographic order on the structure of first  $\Psi$ , then  $X$ . We say  $(M/x : V) \in \{\sigma/\Psi\}$  if, for some  $n$ , the  $n^{\text{th}}$  item from the left in  $\sigma$  is  $M$ , and the  $n^{\text{th}}$  item from the right in  $\Psi$  is  $x : V$ . Almost all cases are homomorphic. Abbreviating  $\theta = \{\sigma/\Psi\}$ , we have

$$\begin{aligned} (M_1 \dots M_n.\text{id})\theta &= (M_1\theta \dots M_n\theta.\text{id}) \\ (V_1, \dots, V_n)\theta &= (V_1\theta, \dots, V_n\theta) \\ (\Pi\Delta.v)\theta &= \Pi(\Delta\theta).(v\theta) \quad (\lambda\hat{\Delta}.R)\theta = \lambda\hat{\Delta}.(R\theta) \\ (\text{type})\theta &= \text{type} \end{aligned}$$

The only interesting case is

$$(x[\rho])\theta = \begin{cases} R\{\rho\theta/\Delta\} & \text{if } (\lambda\hat{\Delta}.R/x : \Pi\Delta.V) \in \theta; \\ x[\rho\theta] & \text{otherwise.} \end{cases}$$

This completes the definition of TinyLF.

## 4 Results

The things we are trying to prove are fairly standard, and so too are the methods for the most part. In preparation for the substitution property, we show that substitutions commute properly. In preparation for the identity property, we show that  $\eta$ -expansions are two-sided units with respect to substitution.

**Lemma 4.1**  $X\{\sigma/\Psi\} = X\{\sigma/\Psi\theta\}$ .

**Proof** By induction on  $\Psi$ , then  $X$ . The interesting case is when  $X = x[\rho]$ , and  $(\lambda\hat{\Delta}.R/x : \Pi\Delta.V) \in \{\sigma/\Psi\}$ . In this case  $(\lambda\hat{\Delta}.R/x : \Pi(\Delta\theta).(V\theta)) \in \{\sigma/\Psi\theta\}$ . We reason that

$$\begin{aligned} & R\{\rho\{\sigma/\Psi\}/\Delta\} \\ &= R\{\rho\{\sigma/\Psi\theta\}/\Delta\} && \text{i.h. on } \rho < x[\rho] \\ &= R\{\rho\{\sigma/\Psi\theta\}/\Delta\theta\} && \text{i.h. on } \Delta < \Psi \end{aligned}$$

■

**Lemma 4.2** *If  $FV(X) \cap \hat{\Psi} = \emptyset$ , then  $X\{\sigma/\Psi\} = X$ .*

**Proof** Straightforward induction. ■

The above lemma is exemplary of the fact that substitution  $\{\sigma/\Psi\}$ , as should be expected from a hereditary substitution, does not really care about the exact structure of the types  $\Psi$ , but only the  $\Pi$ -skeleton of them.

Let  $J$  stand for an arbitrary judgment of the system.

**Lemma 4.3 (Weakening)** *If  $\Gamma, \Gamma' \vdash J$ , then  $\Gamma, \Delta, \Gamma' \vdash J$ .*

**Proof** Straightforward induction. ■

**Lemma 4.4** *If  $(M/x : V) \in \{\sigma/\Psi\}$  and  $\Gamma \vdash \sigma \Leftarrow \Psi$ , then  $\Gamma \vdash M \Leftarrow V\{\sigma/\Psi\}$ .*

**Proof** Straightforward induction. ■

**Lemma 4.5 (Interchange)** *If  $FV(\sigma) \cap \hat{\Delta} = \emptyset$ , then  $X\{\rho/\Delta\}\theta = X\theta\{\rho\theta/\Delta\}$*

**Proof** By induction on first the unordered pair  $(\Delta, \Psi)$ , and subsequently  $X$ . For all the homomorphism cases, it's just  $X$  that gets smaller. This includes the case of  $X = x[\omega]$  where  $x$  is in neither  $\hat{\Delta}$  nor  $\hat{\Psi}$ . The interesting cases are

- $x \in \hat{\Psi}$  and  $(\lambda\hat{\Omega}.R/x : \Pi\Omega.v) \in \{\sigma/\Psi\} = \theta$ . In this case we reason that

$$\begin{aligned} & x[\omega]\{\rho/\Delta\}\theta \\ &= x[\omega\{\rho/\Delta\}]\theta \\ &= R\{\omega\{\rho/\Delta\}\theta/\Omega\} \\ &= R\{\omega\theta\{\rho\theta/\Delta\}/\Omega\} && \text{i.h. on } \rho < x[\rho] \\ &= R\{\rho\theta/\Delta\}\{\omega\theta\{\rho\theta/\Delta\}/\Omega\} && \text{Lemma 4.2} \\ &= R\{\omega\theta/\Omega\}\{\rho\theta/\Delta\} && \text{i.h. on } (\Omega, \Delta) < (\Delta, \Psi) \\ &= x[\omega]\theta\{\rho\theta/\Delta\} \end{aligned}$$

To justify the second induction hypothesis appeal, we need  $FV(\rho\theta) \cap \hat{\Omega} = \emptyset$ , but this is true because the variables  $\hat{\Omega}$  are bound inside  $\sigma$ .

- $x \in \hat{\Delta}$  and  $(\lambda\hat{\Omega}.R/x : \Pi\Omega.v) \in \{\rho/\Delta\}$ . In this case we reason that

$$\begin{aligned}
& x[\omega]\{\rho/\Delta\}\theta \\
&= R\{\omega\{\rho/\Delta\}/\Omega\}\theta \\
&= R\theta\{\omega\{\rho/\Delta\}\theta/\Omega\} && \text{i.h. on } (\Omega, \Psi) < (\Delta, \Psi) \\
&= R\theta\{\omega\theta\{\rho\theta/\Delta\}/\Omega\} && \text{i.h. on } \rho < x[\rho] \\
&= R\{\omega\theta\{\rho\theta/\Delta\}/\Omega\} && \text{Lemma 4.2} \\
&= x[\omega\theta]\{\rho\theta/\Delta\} \\
&= x[\omega]\theta\{\rho\theta/\Delta\}
\end{aligned}$$

■

**Lemma 4.6 (Substitution)** *If  $\Gamma, \Psi, \Gamma' \vdash J$ , and  $\Gamma \vdash \sigma : \Psi$ , then  $\Gamma, \Gamma'\theta \vdash J\theta$ .*

**Proof** By induction on first the  $\Pi$ -skeleton structure of  $\Psi$  (ignoring the identity of all occurring variables and all embedded substitutions) and subsequently the derivation of  $J$ . The interesting cases are:

Case:

$$\frac{\Gamma, \Psi, \Gamma' \vdash M \Leftarrow V\{\rho/\Delta\} \quad \Gamma, \Psi, \Gamma' \vdash \rho \Leftarrow \Delta}{\Gamma, \Psi, \Gamma' \vdash M.\rho \Leftarrow (\Delta, x : V)}$$

Reason that

$$\begin{aligned}
& \Gamma, \Gamma'\theta \vdash \rho\theta \Leftarrow \Delta\theta && \text{i.h.} \\
& \Gamma, \Gamma'\theta \vdash M\theta \Leftarrow V\{\rho/\Delta\}\theta && \text{i.h.} \\
& \Gamma, \Gamma'\theta \vdash M\theta \Leftarrow V\theta\{\rho\theta/\Delta\} && \text{Lemma 4.5} \\
& \Gamma, \Gamma'\theta \vdash M\theta \Leftarrow V\theta\{\rho\theta/\Delta\theta\} && \text{Lemma 4.1}
\end{aligned}$$

and form the derivation

$$\frac{\Gamma, \Gamma'\theta \vdash M\theta \Leftarrow V\theta\{\rho\theta/\Delta\theta\} \quad \Gamma, \Gamma'\theta \vdash \rho\theta \Leftarrow \Delta\theta}{\Gamma, \Gamma'\theta \vdash M\theta.\rho\theta \Leftarrow (\Delta\theta, x : V\theta)}$$

Case:

$$\frac{x : \Pi\Delta.v \in \Gamma, \Psi, \Gamma' \quad \Gamma, \Psi, \Gamma' \vdash \rho : \Delta}{\Gamma, \Psi, \Gamma' \vdash x[\rho] \Rightarrow v\{\rho/\Delta\}}$$

To show:

$$\Gamma, \Gamma'\theta \vdash x[\rho]\theta \Rightarrow v\{\rho/\Delta\}\theta$$

By Lemmas 4.5 and 4.1 it suffices to show

$$\Gamma, \Gamma'\theta \vdash x[\rho]\theta \Rightarrow v\theta\{\rho\theta/\Delta\theta\}$$

and we know already by i.h. that

$$\Gamma, \Gamma'\theta \vdash \rho\theta \Leftarrow \Delta\theta$$

If  $(\lambda\hat{\Delta}.R/x : \Pi\Delta.v) \in \theta$ , then by Lemma 4.4 we know we have a derivation that looks like

$$\frac{\Gamma, \Delta\theta \vdash R \Leftarrow v\theta}{\Gamma \vdash \lambda\hat{\Delta}.R \Leftarrow (\Pi\Delta.v)\theta}$$

and by weakening and i.h. (on  $\Delta\theta < \Psi$ ) and Lemma 4.1 we get

$$\Gamma, \Gamma'\theta \vdash R\{\rho\theta/\Delta\} \Rightarrow v\theta\{\rho\theta/\Delta\theta\}$$

as required. Otherwise, observe that  $(\Gamma, \Gamma')\theta = \Gamma, (\Gamma'\theta)$  by Lemma 4.2, and form the derivation

$$\frac{x : \Pi\Delta\theta.v\theta \in \Gamma, \theta\Gamma' \quad \Gamma, \Gamma'\theta \vdash \rho\theta : \Delta\theta}{\Gamma, \Gamma'\theta \vdash x[\rho\theta] \Rightarrow v\theta\{\rho\theta/\Delta\theta\}}$$

■

Here is how we define  $\eta_V(x)$ , the  $\eta$ -expansion of a variable at a classifier  $V$ , and  $I_\Psi$ , the expanded identity substitution for the context  $\Psi$ .

$$\eta_{\Pi\Psi.v}(x) = \lambda\hat{\Psi}.x[I_\Psi]$$

$$I. = \text{id} \quad I_{\Psi, x:V} = \eta_V(x).I_\Psi$$

**Lemma 4.7 (Unit Laws for  $\eta$ -expansion)**

1. If  $\Gamma \vdash R \Rightarrow v$ , then  $R\{I_\Psi/\Psi\} = R$
2. If  $\Gamma \vdash \rho \Leftarrow \Delta$ , then  $I_\Delta\{\rho/\Delta\} = \rho$
3. If  $\Gamma \vdash \rho \Leftarrow \Delta$ , then  $\rho\{I_\Psi/\Psi\} = \rho$
4. If  $\Gamma \vdash N \Leftarrow V$ , then  $N\{I_\Psi/\Psi\} = N$
5. If  $\Gamma \vdash V \Leftarrow \text{ok}$ , then  $V\{I_\Psi/\Psi\} = V$
6. If  $\Gamma \vdash \Delta \text{ ctx}$ , then  $\Delta\{I_\Psi/\Psi\} = \Delta$

**Proof** By induction.

1. Say  $R = x[\rho]$ . The interesting case is when  $\lambda\hat{\Delta}.x[I_\Delta]/x : \Pi\Delta.v \in \{I_\Psi/\Psi\}$ . Here we want to show  $x[\rho] = x[I_\Delta\{\rho\{I_\Psi/\Psi\}/\Delta\}]$ . By one application of i.h. this is the same as  $x[I_\Delta\{\rho/\Delta\}]$ , and by another we're done.
2. By typing, of  $\rho$  we know that

$$I_\Delta = \eta_{V_1}(x_1) \dots \eta_{V_n}(x_n).\text{id}$$

$$\rho = \lambda\hat{\Psi}_1.R_1, \dots, \lambda\hat{\Psi}_n.R_n.\text{id}$$

supposing that  $V_i = \Pi\Psi_i.v_i$ . We want to show

$$(\eta_{V_1}(x_1) \dots \eta_{V_n}(x_n).\text{id})\{\rho/\Delta\} = \rho$$

so it suffices to show

$$\eta_{V_i}(x_i)\{\rho/\Delta\} = \lambda\hat{\Psi}_i.R_i$$

Let's carry out the eta-expansion on the left:

$$\lambda\hat{\Psi}_i.x_i[I_{\Psi_i}]\{\rho/\Delta\}$$

We know that  $\lambda\hat{\Psi}_i/x : V_i \in \{\rho/\Delta\}$ , so this is

$$\lambda\hat{\Psi}_i.R_i\{I_{\Psi_i}\{\rho/\Delta\}/\Psi_i\}$$

However,  $FV(I_{\Psi_i}) \cap \hat{\Delta} = \emptyset$ , so this is

$$\lambda\hat{\Psi}_i.R_i\{I_{\Psi_i}/\Psi_i\}$$

which by induction hypothesis is

$$\lambda\hat{\Psi}_i.R_i$$

as required.

3. Straightforward.
4. Straightforward.
5. Straightforward.
6. Straightforward.

■

**Lemma 4.8 (Identity)** *Suppose  $\vdash \Gamma \text{ ctx}$ .*

1. *If  $x : V \in \Gamma$ , then  $\Gamma \vdash \eta_V(x) \Leftarrow V$ .*
2. *If  $\Gamma \vdash \Psi \text{ ctx}$ , then  $\Gamma, \Psi \vdash I_\Psi \Leftarrow \Psi$ .*

**Proof** By induction.

1. Form the derivation

$$\frac{\frac{x : \Pi\Psi.v \in \Gamma \quad \Gamma, \Psi \vdash I_\Psi \Leftarrow \Psi}{\Gamma, \Psi \vdash x[I_\Psi] \Leftarrow v\{I_\Psi/\Psi\}}}{\Gamma \vdash \lambda\hat{\Psi}.x[I_\Psi] \Leftarrow \Pi\Psi.(v\{I_\Psi/\Psi\})}$$

which by the previous lemma is as required.

2. The empty context case is immediate. Otherwise, form the derivation

$$\frac{\Gamma, \Psi, x : V \vdash \eta_V(x) \Leftarrow V\{I_\Psi/\Psi\} \quad \Gamma, \Psi \vdash I_\Psi \Leftarrow \Psi}{\Gamma, \Psi, x : V \vdash \eta_V(x).I_\Psi \Leftarrow \Psi, x : V}$$

from uses of the i.h. and previous lemma.

■