

Free Logic and Category Theory in Isabelle/HOL: Experiments

Christoph Benzmlüller and Dana Scott

March 1, 2016

Abstract

An embedding of free logic in classical higher-order logic is presented which has been formalized in Isabelle/HOL. Subsequently this work has been utilized as a foundation for the formalization of Peter Freyd’s axiomatic category theory in Isabelle/HOL. Experiments with automated theorem provers integrated with Isabelle/HOL have been carried out, which revealed a previously unknown redundancy in Freyd’s axiom system.

1 Free Logic

Motivated by problems and shortcomings in the handling of improper descriptions in the works of Russell, Frege and Hilbert-Bernays, the second author has proposed an alternative solution in his 1967 paper *Existence and Description in Formal Logic* [1].

2 Free Logic in HOL

In this section We present an embedding of the second authors *Free Logic* in Isabelle/HOL.

type-synonym $\sigma = \text{bool}$ — the type for Booleans

consts $f\text{-}A :: 'a \Rightarrow \sigma$ (\mathcal{A})

consts $f\text{-}star :: 'a$ (\star)

axiomatization where $f\text{-}star\text{-}axiom: \neg \mathcal{A}(\star)$

Negation and implication in free logic are mapped to negation in HOL.

abbreviation $f\text{-}not :: \sigma \Rightarrow \sigma$ (\neg [58] 59)

where $\neg \varphi \equiv \neg \varphi$

abbreviation $f\text{-}implies :: \sigma \Rightarrow \sigma \Rightarrow \sigma$ (**infixr** \rightarrow 49)

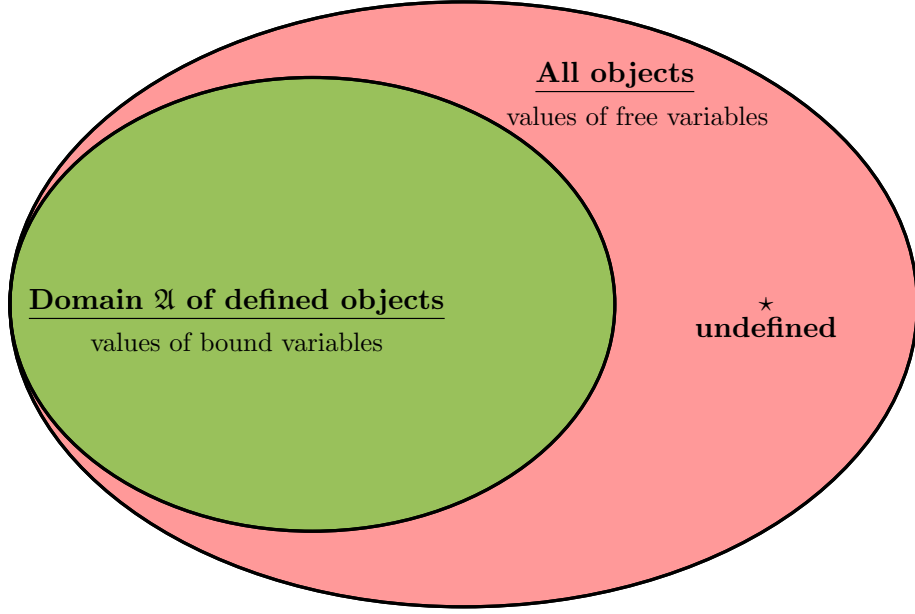


Figure 1: Scott's Free Logic

where $\varphi \rightarrow \psi \equiv \varphi \longrightarrow \psi$

Universal quantification in free logic is restricted to the domain of existing objects

abbreviation $f\text{-all} :: ('a \Rightarrow \sigma) \Rightarrow \sigma \ (\forall)$

where $\forall \Phi \equiv \forall x. \mathcal{A}(x) \longrightarrow \Phi(x)$

abbreviation $f\text{-all-bind} :: ('a \Rightarrow \sigma) \Rightarrow \sigma \ (\text{binder } \forall \ [8] \ 9)$

where $\forall x. \varphi(x) \equiv \forall \varphi$

abbreviation $f\text{-that} :: ('a \Rightarrow \sigma) \Rightarrow 'a \ (\mathbf{I})$

where $\mathbf{I} \Phi \equiv \text{if } \exists x. \mathcal{A}(x) \wedge \Phi(x) \wedge (\forall y. (\mathcal{A}(y) \wedge \Phi(y)) \longrightarrow (y = x))$
 then $\text{THE } x. \mathcal{A}(x) \wedge \Phi(x)$
 else \star

abbreviation $f\text{-that-b} :: ('a \Rightarrow \sigma) \Rightarrow 'a \ (\text{binder } \mathbf{I} \ [8] \ 9)$

where $\mathbf{I}x. \varphi(x) \equiv \mathbf{I}(\varphi)$

abbreviation $f\text{-or} :: \sigma \Rightarrow \sigma \Rightarrow \sigma \ (\text{infixr } \vee \ 51)$

where $\varphi \vee \psi \equiv (\neg \varphi) \rightarrow \psi$

abbreviation $f\text{-and} :: \sigma \Rightarrow \sigma \Rightarrow \sigma \ (\text{infixr } \wedge \ 52)$

where $\varphi \wedge \psi \equiv \neg(\neg \varphi \vee \neg \psi)$

abbreviation $f\text{-equiv} :: \sigma \Rightarrow \sigma \Rightarrow \sigma \ (\text{infixr } \leftrightarrow \ 50)$

where $\varphi \leftrightarrow \psi \equiv (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$

abbreviation $f\text{-equals} :: 'a \Rightarrow 'a \Rightarrow \sigma \ (\text{infixr } = \ 56)$

where $x = y \equiv x = y$

abbreviation $f\text{-exi} :: ('a \Rightarrow \sigma) \Rightarrow \sigma \ (\exists)$

where $\exists \Phi \equiv \neg \forall (\lambda y. \neg (\Phi \ y))$
abbreviation $f\text{-}exi\text{-}b :: ('a \Rightarrow \sigma) \Rightarrow \sigma$ (**binder** \exists $[8]9$)
where $\exists x. \varphi(x) \equiv \exists \varphi$

3 Some Introductory Tests

— See Scott, Existence and Description in Formal Logic, 1967, pages 183-184

typeddecl ι — the type for individuals
consts $f\text{-}r :: \iota \Rightarrow \iota \Rightarrow \sigma$ (**infixr** r 70)

lemma $x \ r \ x \rightarrow x \ r \ x$ **by** *simp*

lemma $\exists y. y \ r \ y \rightarrow y \ r \ y$ **nitpick** **oops**

lemma $(x \ r \ x \rightarrow x \ r \ x) \rightarrow (\exists y. y \ r \ y \rightarrow y \ r \ y)$ **nitpick** **oops**

lemma $((x \ r \ x \rightarrow x \ r \ x) \wedge (\exists y::\iota. y = y)) \rightarrow (\exists y. y \ r \ y \rightarrow y \ r \ y)$ **by** *simp*

— See Scott 1967, page 185

lemma $S1\text{-}inst : (\forall x. \Phi(x) \rightarrow \Psi(x)) \rightarrow ((\forall x. \Phi(x)) \rightarrow (\forall x. \Psi(x)))$ **by** *auto*

lemma $S2 : \forall y. \exists x. x = y$ **by** *auto*

lemma $S3 : \alpha = \alpha$ **by** *auto*

lemma $S4\text{-}inst : (\Phi(\alpha) \wedge (\alpha = \beta)) \rightarrow \Phi(\beta)$ **by** *auto*

lemma $UI\text{-}inst : ((\forall x. \Phi(x)) \wedge (\exists x. x = \alpha)) \rightarrow \Phi(\alpha)$ **by** *auto*

lemma $UI\text{-}test : (\forall x. \Phi(x)) \rightarrow \Phi(\alpha)$ **nitpick** [*user-axioms*] **oops** — Counter-model

lemma $UI\text{-}cor1 : \forall y. ((\forall x. \Phi(x)) \rightarrow \Phi(y))$ **by** *auto*

lemma $UI\text{-}cor2 : \forall y. ((\forall x. \neg(x = y)) \rightarrow \neg(y = y))$ **by** *auto*

lemma $UI\text{-}cor3 : \forall y. ((y = y) \rightarrow (\exists x. x = y))$ **by** *auto*

lemma $UI\text{-}cor4 : (\forall y. y = y) \rightarrow (\forall y. \exists x. x = y)$ **by** *simp*

lemma $(\exists x. x = \alpha) \longrightarrow \mathcal{A}(\alpha)$ **by** *simp*

lemma $I1\text{-}inst : \forall y. ((y = (\mathbf{I}x. \Phi(x))) \leftrightarrow (\forall x. ((x = y) \leftrightarrow \Phi(x))))$ **by** (*smt* *f-star-axiom the-equality*)

abbreviation $star (\otimes)$ **where** $\otimes \equiv \mathbf{I}y. \neg (y = y)$

lemma $test : \otimes = \star$ **by** *simp*

lemma $I2\text{-}inst : \neg(\exists y. y = (\mathbf{I}x. \Phi(x))) \rightarrow (\otimes = (\mathbf{I}x. \Phi(x)))$ **by** (*metis* (*no-types*, *lifting*) *the-equality*)

lemma *Ext-inst* : $(\forall x. \Phi(x) \leftrightarrow \Psi(x)) \rightarrow ((\mathbf{I}x. \Phi(x)) = (\mathbf{I}x. \Psi(x)))$ **by** (*smt the1-equality*)

lemma *I3* : $(\otimes = \alpha \vee \otimes = \beta) \rightarrow \neg(\alpha \mathbf{r} \beta)$ **nitpick** [*user-axioms*] **oops**

lemma *Russel-inst* :
 $((\otimes = \alpha \vee \otimes = \beta) \rightarrow \neg(\alpha \mathbf{r} \beta))$
 \rightarrow
 $((\alpha \mathbf{r} (\mathbf{I}x. \Phi(x))) \leftrightarrow (\exists y. ((\forall x. ((x = y) \leftrightarrow \Phi(x))) \wedge (\alpha \mathbf{r} y))))$
nitpick [*user-axioms*] **oops**

lemma $\neg(\exists x. (x = (\mathbf{I}y. \neg(y = y))))$ **using** *f-star-axiom* **by** *auto*
lemma $(\exists x. x = a) \rightarrow \mathcal{A}(a)$ **by** *simp*

consts *ca::'a cb::'a*
axiomatization where *ax1*: $\mathcal{A}(ca) \wedge \mathcal{A}(cb) \wedge \neg(ca = cb) \wedge \neg(ca = \otimes) \wedge \neg(cb = \otimes)$
lemma *test2*: $\otimes = (\mathbf{I}(\lambda x. x = ca \vee x = cb))$ **by** (*metis ax1*)

end
theory *Freyd* **imports** *FreeHOL*
begin

typedecl *e* — raw type of morphisms
abbreviation *Definedness* :: $e \Rightarrow \text{bool}$ (*D*-[8]60)
where $D\ x \equiv \mathcal{A}\ x$
abbreviation *OrdinaryEquality* :: $e \Rightarrow e \Rightarrow \text{bool}$ (**infix**≈60)
where $x \approx y \equiv ((D\ x) \leftrightarrow (D\ y)) \wedge x = y$

consts *source* :: $e \Rightarrow e$ (\square - [108]109)
target :: $e \Rightarrow e$ ($-\square$ [110]111)
composition :: $e \Rightarrow e \Rightarrow e$ (**infix** · 110)

axiomatization *FreydsAxioms* **where**
A1: $(D\ x \cdot y) \leftrightarrow ((x \square) \approx (\square y))$ **and**

A2b: $\square(x \square) \approx \square x$ **and**
A3a: $(\square x) \cdot x \approx x$ **and**

A3b: $x \cdot (x\Box) \approx x$ **and**
A4a: $\Box(x \cdot y) \approx \Box(x \cdot (\Box y))$ **and**
A4b: $(x \cdot y)\Box \approx ((x\Box) \cdot y)\Box$ **and**
A5: $x \cdot (y \cdot z) \approx (x \cdot y) \cdot z$

lemma L1: $(\Box\Box x) \cdot ((\Box x) \cdot x) \approx ((\Box\Box x) \cdot (\Box x)) \cdot x$ **using A5 by metis**
lemma L2: $(\Box\Box x) \cdot x \approx ((\Box\Box x) \cdot (\Box x)) \cdot x$ **using L1 A3a by metis**
lemma L3: $(\Box\Box x) \cdot x \approx (\Box x) \cdot x$ **using L2 A3a by metis**
lemma L4: $(\Box\Box x) \cdot x \approx x$ **using L3 A3a by metis**
lemma L5: $\Box((\Box\Box x) \cdot x) \approx \Box((\Box\Box x) \cdot (\Box x))$ **using A4a by auto**
lemma L6: $\Box((\Box\Box x) \cdot x) \approx \Box\Box x$ **using L5 A3a by metis**
lemma L7: $\Box\Box(x\Box) \approx \Box(\Box\Box(x\Box)) \cdot (x\Box)$ **using L6 by auto**
lemma L8: $\Box\Box(x\Box) \approx \Box(x\Box)$ **using L4 L7 by metis**
lemma L9: $\Box\Box(x\Box) \approx \Box x$ **using A2b L8 by metis**
lemma L10: $\Box\Box x \approx \Box x$ **using A2b L9 by metis**
lemma L11: $\Box\Box((\Box x)\Box) \approx \Box\Box(x\Box)$ **using A2b L10 by metis**
lemma L12: $\Box\Box((\Box x)\Box) \approx \Box x$ **using L11 L9 by metis**
lemma L13: $(\Box\Box((\Box x)\Box)) \cdot ((\Box x)\Box) \approx ((\Box x)\Box)$ **using L4 by auto**
lemma L14: $(\Box x) \cdot ((\Box x)\Box) \approx (\Box x)\Box$ **using L12 L13 by metis**
lemma LM10: $(\Box x)\Box \approx (\Box x) \cdot ((\Box x)\Box)$ **using L14 by auto**
lemma A2a: $(\Box x)\Box \approx \Box x$ **using A3b LM10 by metis**

abbreviation DirectedEquality :: $e \Rightarrow e \Rightarrow \text{bool}$ (**infix** \gtrsim 60)
where $x \gtrsim y \equiv ((D\ x) \rightarrow (D\ y)) \wedge x = y$

lemma L1-13: $((\Box(x \cdot y)) \approx (\Box(x \cdot (\Box y)))) \leftrightarrow ((\Box(x \cdot y)) \gtrsim \Box x)$
by (metis A1 A2a A3a)

lemma $(\exists x. e \approx (\Box x)) \leftrightarrow (\exists x. e \approx (x\Box))$
by (metis A1 A2b A3b)
lemma $(\exists x. e \approx (x\Box)) \leftrightarrow e \approx (\Box e)$
by (metis A1 A2b A3a A3b)
lemma $e \approx (\Box e) \leftrightarrow e \approx (e\Box)$
by (metis A1 A2b A3a A3b A4a)
lemma $e \approx (e\Box) \leftrightarrow (\forall x. e \cdot x \gtrsim x)$
by (metis A1 A2b A3a A3b A4a)
lemma $(\forall x. e \cdot x \gtrsim x) \leftrightarrow (\forall x. x \cdot e \gtrsim x)$
by (metis A1 A2b A3a A3b)

abbreviation IdentityMorphism :: $e \Rightarrow \text{bool}$ (**IdM**- [8]60) **where** $\text{IdM } x \equiv x \approx (\Box x)$

lemma $(\text{IdM } e \leftrightarrow (\exists x. e \approx (\Box x))) \wedge$
 $(\text{IdM } e \leftrightarrow (\exists x. e \approx (x\Box))) \wedge$
 $(\text{IdM } e \leftrightarrow e \approx (\Box e)) \wedge$

$$\begin{aligned}
& (IdM\ e \leftrightarrow e \approx (e\Box)) \wedge \\
& (IdM\ e \leftrightarrow (\forall x. e \cdot x \overset{\sim}{\approx} x)) \wedge \\
& (IdM\ e \leftrightarrow (\forall x. x \cdot e \overset{\sim}{\approx} x)) \\
\text{by } & (smt\ A1\ A2a\ A3a\ A3b) \\
\text{end}
\end{aligned}$$

References

- [1] D. Scott. Existence and description in formal logic. In R. Schoenman, editor, *Bertrand Russell: Philosopher of the Century*, pages 181–200. George Allen & Unwin, London, 1967. (Reprinted with additions in: *Philosophical Application of Free Logic*, edited by K. Lambert. Oxford University Press, 1991, pp. 28 - 48).