Formal typesytem description

Ferdinand van Walree(3874389) and Matthew Swart(5597250)

November 29, 2015

0.1 Introduction

This document serves as a formal description of the typesystem developed by Ferdinand van Walree en Matthew Swart.

0.2 Typesystem

0.3 Ty

The Ty is a datatype that contains all the information about the type itself.

$$\frac{}{\Gamma \Vdash \text{Program p l} : \text{(Ty Prog (just l) Nothing Nothing)}}\ \text{(ty-prog)}$$

$$\frac{}{\Gamma \Vdash \text{Program p l} : \text{Prog}}\ \text{(tyCons-prog)}$$

$$\frac{}{\Gamma \Vdash \text{Interpreter i l m} : \text{(Ty Interp (Just l) Nothing (Just m)}}\ \text{(ty-interp)}$$

$$\frac{}{\Gamma \Vdash \text{Interper i l m} : \text{Interp}}\ \text{(tyCons-interp)}$$

$$\frac{}{\Gamma \Vdash \text{Compiler c l1 l2 m} : \text{Comp (Just l1) (Just l2) (Just m)}}\ \text{(ty-comp)}$$

$$\frac{}{\Gamma \Vdash \text{Compilerc l1 l2 ml} : \text{Comp}}\ \text{(tyCons-comp)}$$

$$\frac{}{\text{Platform m} : \text{Ty PlatF Nothing Nothing (Just m)}}\ \text{(ty-platf)}$$

$$\frac{}{\Gamma \Vdash \text{Platform m} : \text{Platform}}\ \text{(tyCons-platf)}$$

$$\frac{}{\Gamma \Vdash \text{Execute d1 d2} : \text{d2}}\ \text{(ty-execute)}$$

$$\frac{}{\Gamma \Vdash \text{Execute p l} : \text{Execute}}\ \text{(tyCons-execute)}$$

$$\frac{\Gamma \Vdash \text{ty: Ty ty2: Ty}}{\Gamma \Vdash \text{Compiler d1 d2: translate}}\ \text{(ty-compile)}$$

$$\frac{}{\Gamma \Vdash \text{Compile d1 d2} : \text{Compiled}}\ \text{(tyCons-compile)}$$

$$\frac{}{\Gamma \Vdash \text{Prog} : \text{TyCons}}\ \text{(t-ProgTyCons)}$$

$$\frac{}{\Gamma \Vdash \text{Interp} : \text{TyCons}}\ \text{(t-InterpTyCons)}$$

$$\frac{}{\Gamma \Vdash \text{Comp} : \text{TyCons}}\ \text{(t-CompTyCons)}$$

$$\frac{}{\Gamma \Vdash \text{PlatF} : \text{TyCons}}\ \text{(t-PlatfTyCons)}$$

$$\frac{}{\Gamma \Vdash \text{Prog} : \text{TyCons}}\ \text{(t-ProgTyCons)}$$

0.4 checkRunnable

The checkRunnable is used to refuse a platform that is being executed or compiled.

$$\frac{\Gamma \Vdash \text{t1} : \text{Platf t2} : \text{TyCons}}{\Gamma \Vdash \text{ill-typed}} \text{ (checkrunnable-False)}$$

0.5 checkComp

The checkComp is used to only accept a compiler when its type is Comp, because a Compile needs a compiler to compile.

$$\frac{\Gamma\ (\text{t} : \text{Compile}) \Vdash \text{t1} : \text{TyCons t2} : \neg\text{Comp}}{\Gamma \Vdash \text{ill-typed}} \text{ (checkComp-False)}$$

0.6 checkExeInComp

The checkExeInComp is used to refuse when a executed is in the compiler.

$$\frac{\dfrac{\Gamma \Vdash \text{t1} : \text{TyCons}\ \Gamma \Vdash \text{t2} : \text{Executed}}{\Gamma \Vdash \text{match-eq-t1-t2} : \text{True}}}{\Gamma \Vdash \text{ill-typed}} \text{ (checkExeInComp-True)}$$

0.7 checkExeOrComp

The checkExeOrComp is used to refuse an executed or compiled in an Execute.

$$\frac{\Gamma \Vdash \text{t1} : \text{TyCons } (\Gamma \Vdash \text{t1} : \text{Executed} \bigvee \Gamma \Vdash \text{t2} : \text{Compiled})}{\Gamma \Vdash \text{ill-typed}} \text{ (checkExeOrComp-false)}$$

0.8 checkFramework

The checkFramework is used to only accept in the Execute a Interpeter or a Platform.

$$\frac{\Gamma \Vdash \text{t1} : \text{TyCons } \Gamma \Vdash \text{t2} : \text{Framework}}{\dfrac{\text{match-nq-t1-t2}}{\Gamma \Vdash \text{ill-typed}} \text{ (checkFramework-False)}}$$

0.9 Match

The match is used to pattern match on what is possible and what is not.

$$\frac{\Gamma \Vdash \text{Prog} : \text{TyCons } \Gamma \Vdash \text{Runnable} : \text{TyCons}}{\Gamma \Vdash \text{True} : \text{Bool}} \text{ (match-prog-runnable)}$$

$$\frac{\Gamma \Vdash \text{Interp} : \text{TyCons } \Gamma \Vdash \text{Runnable} : \text{TyCons}}{\Gamma \Vdash \text{True} : \text{Bool}} \text{ (match-interp-runnable)}$$

$$\frac{\Gamma \Vdash \text{Comp} : \text{TyCons } \Gamma \Vdash \text{Runnable} : \text{TyCons}}{\Gamma \Vdash \text{True} : \text{Bool}} \text{ (match-comp-runnable)}$$

$$\frac{\Gamma \Vdash \text{Interp} : \text{TyCons } \Gamma \Vdash \text{Framework} : \text{TyCons}}{\Gamma \Vdash \text{True} : \text{Bool}} \text{ (match-interp-framework)}$$

$$\frac{\Gamma \Vdash \text{PlatF} : \text{TyCons}\ \Gamma \Vdash \text{Framework} : \text{TyCons}}{\Gamma \Vdash \text{True} : \text{Bool}}\ \text{(match-platf-framework)}$$

$$\frac{\Gamma \Vdash \text{t1:}\ \text{TyCons}\ \Gamma \Vdash \text{t2:}\ \text{TyCons}}{\dfrac{\Gamma \Vdash \text{t1} \equiv \text{t2} : \text{Bool}}{\Gamma \Vdash \text{True} : \text{Bool}}}\ \text{(match-eq-t1-t2)}$$

$$\frac{\Gamma \Vdash \text{t1:}\ \text{TyCons}\ \Gamma \Vdash \text{t2:}\ \text{TyCons}}{\dfrac{\Gamma \Vdash \text{t1} \neq \text{t2} : \text{Bool}}{\Gamma \Vdash \text{False} : \text{Bool}}}\ \text{(match-nq-t1-t2)}$$

0.10 Match info

matchInfo is used to compare two values, so we can tell if a source, platform or target language are equal.

$$\frac{\text{matchInfo-eq} \bigvee \text{matchInfo-gen}}{\Gamma \Vdash \text{True} : \text{Bool}}\ \text{(matchInfo)}$$

$$\frac{\Gamma \Vdash \text{just i} : \text{Maybe Ident}\ \Gamma \Vdash \text{just j}\ \Gamma \Vdash \text{Maybe Ident}}{\dfrac{\Gamma \Vdash \text{i} \equiv \text{j} : \text{Bool}}{\Gamma \Vdash \text{True} : \text{Bool}}}\ \text{(matchInfo-eq)}$$

$$\frac{\Gamma \Vdash \text{just i} : \text{Maybe Ident}\ \Gamma \Vdash \text{just j}\ \Gamma \Vdash \text{Maybe Ident}}{\dfrac{\Gamma \Vdash \text{i} \neq \text{j} : \text{Bool}}{\Gamma \Vdash \text{False} : \text{Bool}}}\ \text{(matchInfo-nq)}$$

$$\frac{\Gamma\ \Gamma \Vdash \text{m1} : \text{Maybe Ident}\ \Gamma \Vdash \text{m2:}\ \text{Maybe Ident}}{\Gamma \Vdash \text{True} : \text{Bool}}\ \text{(matchInfo-gen)}$$

0.11 Translate

The translate is used to get the correct Ty for the Compile.

$$\frac{\Gamma \Vdash (\text{Ty Prog s1 t1 m1})\ \Gamma \Vdash (\text{Ty comp s2 t2 m2})}{\Gamma \Vdash \text{Ty prog t2 t1 m1}}\ \text{(translate-prog-comp)}$$

$$\frac{\Gamma \Vdash (\text{Ty Interp s2 t2 m2})\ \Gamma \Vdash (\text{Ty comp s2 t2 m2})}{\Gamma \Vdash \text{Ty prog s1 t1 t2}}\ \text{(translate-interp-comp)}$$

$$\frac{\Gamma \Vdash (\text{Ty comp s1 t1 m1})\ \Gamma \Vdash (\text{Ty comp s2 t2 m2})}{\Gamma \Vdash \text{Ty comp s1 t1 t2}}\ \text{(translate-comp-comp)}$$

0.12 Checkifmatch

In the checkmatch we check on each type combination if it's being executed, compiled or interpreted on the same language as the source, target or platform. We use the matchInfo for this.

$$\frac{\Gamma \Vdash (\text{Ty Comp s1 t1 m1})\ \Gamma \Vdash (\text{Ty Interp s2 t2 m2})}{\dfrac{\Gamma \Vdash \text{matchinfo-False}}{\Gamma \Vdash \text{ill-typed}}}\ \text{(checkifmatch-prog-interp-ill)}$$

$$\frac{\Gamma \Vdash \text{ty1: Ty}\ \Gamma \Vdash (\text{Ty Interp s2 t2 m2})}{\dfrac{\Gamma \Vdash \text{matchinfo-False}}{\Gamma \Vdash \text{ill-typed}}}\ \text{(checkifmatch-unknown-interp-ill)}$$

$$\frac{\Gamma \Vdash \text{(Ty Prog s1 t1 m1)}\ \Gamma \Vdash \text{(Ty PlatF s2 t2 m2)}}{\dfrac{\Gamma \Vdash \text{matchinfo-False}}{\Gamma \Vdash \text{ill-typed}}}\ \text{(checkifmatch-prog-platf-ill)}$$

$$\frac{\Gamma \Vdash \text{ty1 : Ty}\ \Gamma \Vdash \text{(Ty PlatF s2 t2 m2)}}{\dfrac{\Gamma \Vdash \text{matchinfo-False}}{\Gamma \Vdash \text{ill-typed}}}\ \text{(checkifmatch-unknown-platf-ill)}$$

$$\frac{\Gamma \Vdash \text{(Ty Prog s1 t1 m1)}\ \Gamma \Vdash \text{(Ty Comp s2 t2 m2)}}{\dfrac{\Gamma \Vdash \text{matchinfo-False}}{\Gamma \Vdash \text{ill-typed}}}\ \text{(checkifmatch-prog-comp-ill)}$$

$$\frac{\Gamma \Vdash \text{ty1 : Ty}\ \Gamma \Vdash \text{(Ty comp s2 t2 m2)}}{\dfrac{\Gamma \Vdash \text{matchinfo-False}}{\Gamma \Vdash \text{ill-typed}}}\ \text{(checkifmatch-unkown-comp-ill)}$$