3. Type Systems

We will consider the problem of type assignment and type checking for the object calculus.

Just as we have type-free and typed lambda calculus, so we also have type-free and typed object calculus.

3.1. Typed Lambda Calculus

Recall that we can assign a type to every type free lambda term, if we make some assumptions about the type of its variable symbols.

This is no different from asserting a ***type declaration*** about those variable.

Type Judgments

We make ***type judgments*** of the form

$$x_1 : \tau_1, .., x_n : \tau_n \supset \tau$$

Meaning ***in the type environment that x_1 has type τ_1 and …, and x_n has type τ_n then τ is a (well-formed) type.***

Value Judgments

We also make ***value judgments*** of the form

$$x_1 : \tau_1, .., x_n : \tau_n \supset t : \tau$$

Meaning, ***if*** x_1 ***has type*** τ_1 ***and …, and*** x_n ***has type*** τ_n ***then the lambda term t has type*** τ.

Rules of Λ

The rules are just like proof rules in logic, and we use them this way.
They inductively follow the construction of terms.

We use A, B, C, to denote type variables, and E, E’, E’’ to denote type environments (declarations)

(1. Env ϕ)

$$\frac{}{\phi \supset \diamond}$$

(2. Env x)

$$\frac{E \supset A, \;\; x \notin Dom(E)}{E, x{:}A \supset \diamond}$$

(3. Val x)

$$\frac{E', x{:}A, E'' \supset \diamond}{E', x{:}A, E'' \supset x{:}A}$$

(4. Type Arrow)

$$\frac{E \supset A \quad E \supset B}{E \supset A \to B}$$

(5. Val Fun)

$$\frac{E, x{:}A \supset b{:}B}{E \supset \lambda(x{:}A)b : A \to B}$$

(6. Val Appl)

$$\frac{E \supset b : A \to B \quad E \supset a : A}{E \supset b(a) : B}$$

Example

Using the above 6 rules we can formally prove the type judgment

$$x{:}A \supset A \to A$$

and the value judgment

$$x{:}A \supset \lambda(x{:}A).x : A \to A$$

(Exercise!)

Notice we have now expanded our language of lambda terms to include type assertions e.g.

$$\lambda(x:A).x$$

instead of

$$\lambda(x).x$$

We will not take the trouble to redefine the language again, since this should be obvious.

3.2. Extending to Typed Objects

Since the lambda calculus is embedded in the object calculus, we just need to extend our type and value judgments to include all objects and object types.

But what **is** an object type?

(7. Type Object)

$$\frac{E \supset B_i \quad \forall i \in \{1,.., n\}}{E \supset [l_i : B_i^{\ i \in \{1,.., n\}}]}$$

(8. Val Object) (where $A = [l_i : B_i^{\ i \in 1,.., n}]$)

$$\frac{E, x_i{:}A \supset b_i : B_i \quad \forall i \in \{1,.., n\}}{E \supset [l_i : \zeta(x_i{:}A)\ b_i^{\ i \in 1,.., n}]{:}A}$$

(9. Val Select)

$$\frac{E \supset a:[l_i : B_i{}^{i\in \{1,..,n\}}] \quad j\in\{1,..,n\}}{E \supset a.l_j : B_j}$$

(10. Val Update) (where $A = [l_i : B_i{}^{i\in 1,..,n}]$)

$$\frac{E \supset a:A \qquad E, x:A \supset b : B_j \quad j\in\{1,..,n\}}{E \supset a.l_j \Leftarrow \zeta(x:A)\, b : A}$$

We call this type calculus with rules 1...10

FOB_1

3.3. Properties of $\mathbf{FOB_1}$

Unique Types Theorem

If $E \supset a : A$ and $E \supset a : B$

are derivable in $\mathbf{FOB_1}$ then $A = B$.

Proof

Easy induction on the buildup of value judgments. (Exercise)

Operational Semantics of Typed Object Terms

We generalize the weak reduction relation to typed terms

(Red Object)

(where $v = [l_i : \zeta(x_i{:}A)\, b_i{}^{\,i\in 1,..,n}]$)

$$\frac{}{\supset v \Rightarrow^{WR} v}$$

(Red Select)

(where $v' = [l_i : \zeta(x_i{:}A)\ b_i\ \{x_i\}^{i \in 1,.., n}]$)

$$\frac{\supset a \Rightarrow^{WR} v' \qquad \supset b_j\{x_i \to v'\} \Rightarrow^{WR} v}{\supset a.l_j \Rightarrow^{WR} v}$$

(Red Update)

$$\frac{\supset a \Rightarrow^{WR} [l_i : \zeta(x_i{:}A_i)\ b_i\ ^{i \in 1,.., n}] \qquad j \in \{1,.., n\}}{\supset a.l_j \Leftarrow \zeta(x{:}A)\ b \Rightarrow^{WR} [l_j : \zeta(x{:}A_j)\ b,\ l_i : \zeta(x_i{:}A_i)\ b_i\ ^{i \in \{1,.., n\}-\{j\}}]}$$

Properties (Continued)

Subject Reduction Theorem

Let c be a closed typed object term and v be a result. If

$$\supset c \Rightarrow^{WR} v$$

and in $\mathbf{FOB_1}$

$$\phi \supset c:C$$

then also in $\mathbf{FOB_1}$

$$\phi \supset v:C$$

3.4. Subtyping

A characteristic of OO languages is that an object can emulate another object that has fewer methods.

We call this ***subsumption***: an object can subsume another object with a more limited protocol.

We will introduce a subtype relation on types.

Then we will allow an object to belong to any supertype of its type.

Hence it can subsume objects of the supertype.

Subtype Judgments

We introduce new ***subtype judgements*** of the form

$$E \supset A < B$$

Meaning ***in the type environment E, type A is a subtype of type B.***

(11. Sub Reflex)

$$\frac{E \supset A}{E \supset A<A}$$

(12. Val Subsumpt)

$$\frac{E \supset a : A \quad E \supset A<B}{E \supset a : B}$$

(13. Sub Trans)

$$\frac{E \supset A<B \quad E \supset B<C}{E \supset A<C}$$

An Object Type

We will assume a ***type for all objects*** (called Top) which subsumes every other type.This is like the type **`Object`** in Java.

(14. Type Top)

$$\frac{E \supset \diamond}{E \supset \mathrm{Top}}$$

(15. Sub Top)

$$\frac{E \supset A}{E \supset A < \mathrm{Top}}$$

(16. Sub Arrow)

$$\frac{E \supset A' < A \quad E \supset B < B'}{E \supset A \rightarrow B < A' \rightarrow B'}$$

(17. Sub Object)

$$\frac{E \supset B_i \qquad \forall i \in \{1,.., n+m\}}{E \supset [l_i : B_i{}^{i \in \{1,.., n+m\}}] < [l_i : B_i{}^{i \in \{1,.., n\}}]}$$

3.5. Properties of $\mathbf{OB}_{1<}$

Let us call the system of rules 1-3,7-15,17 (observe lambda terms and rules are omitted!) $\mathbf{OB}_{1<}$. Obviously by subsumption, we have lost the unique types property. But we have an acceptable weaker property of ***minimum*** (or ***most specific***) ***types***.

Minimum Types

<u>Minimum Types Theorem for $\mathbf{OB_{1<}}$</u>

If $E \supset a : A$ is derivable in $\mathbf{OB_{1<}}$

then there exists B (the minimum type of a)

such that $E \supset a : B$ is derivable in $\mathbf{OB_{1<}}$

and for any A’ if $E \supset a : A'$ then

$E \supset B < A'$.

We prove the Minimum Types Theorem by introducing an alternative proof calculus where we remove the Val Subsumpt rule 12 (which destroys the unique types property) and modify the Val Object and Val Update rules 8, 10.

(18. Val Min Object)

(where $A = [l_i : B_i \ ^{i \in 1,..,n}]$)

$$\frac{E, x_i{:}A \supset b_i : B'_i \quad \varphi \supset B'_i < B_i \quad \forall i \in \{1,..,n\}}{E \supset [l_i : \zeta(x_i{:}A)\, b_i \ ^{i \in 1,..,n}]{:}A}$$

(19. Val Min Update)

(where $A = [l_i : B_i \ ^{i \in 1,.., n}]$)

$\underline{E \supset a:A' \quad \varphi \supset A' < A \quad E, x:A \supset b : B'_j}$

$\underline{\varphi \supset B'_j < B_j \quad j \in \{1,.., n\}}$

$E \supset a.l_j \Leftarrow \zeta(x:A)\, b : A$

Let us call the system of rules 1-3,7,9,11,15,17-19 $\mathbf{MinOB_{1<}}$.

Proposition 1

(**MinOB$_{1<}$** typings are **OB$_{1<}$** typings)

If $E \supset a : A$ is derivable in **MinOB$_{1<}$**

then $E \supset a : A$ is derivable in **OB$_{1<}$**

Proposition 2.

If $E \supset a : A$ and $E \supset a : B$

are derivable in **MinOB$_{1<}$** then $A = B$.

Proposition 3.

(**$\mathbf{MinOB_{1<}}$** has smaller types than **$\mathbf{OB_{1<}}$**)

If $E \supset a : A$ is derivable in **$\mathbf{OB_{1<}}$**

then $E \supset a : B$ is derivable in **$\mathbf{MinOB_{1<}}$**

for some type B such that

$E \supset B < A$ is derivable (in either system)

Proof. (Of the Minimum Types Theorem)

Suppose $E \supset a : A$ is derivable in $\mathbf{OB}_{1<}$

By Prop 3, $E \supset a : B$ is derivable in $\mathbf{MinOB}_{1<}$

For some B s.t. $E \supset B < A$. By Prop 1.

$E \supset a : B$ is also derivable in $\mathbf{OB}_{1<}$.

We will show that B is the minimum type of the object a in $\mathbf{OB}_{1<}$.

Suppose E $\supset$ a : C is derivable in $\mathbf{OB_{1<}}$ for any other type C, then we must show that C is a supertype of B. By Prop. 3, E $\supset$ a : C’ is derivable in $\mathbf{MinOB_{1<}}$ for some C’ such that E $\supset$ C’ < C. By Proposition 2, C’=B.

So E $\supset$ B < C, i.e. C is indeed a supertype of B. Since C was arbitrarily chosen then B is the minimum type of object a.

Subject Reduction

Subject Reduction Theorem for $\mathbf{OB}_{1<}$

Let c be a closed typed object term and v be a result. If

$$\supset c \Rightarrow^{WR} v$$

and in $\mathbf{OB}_{1<}$

$$\phi \supset c{:}C$$

then also in $\mathbf{OB}_{1<}$

$$\phi \supset v{:}C$$