

Mix-in composition

Julien Wetterwald

CS94SI

Multiple inheritance vs mix-in composition

- In multiple inheritance, a super call invokes a method in one of the superclasses of the calling class
- With traits, the invoked method is found according to a linearization of the classes and traits that are mixed into a class

Linearization

Let C be a class with template C_1 with ... with C_n .

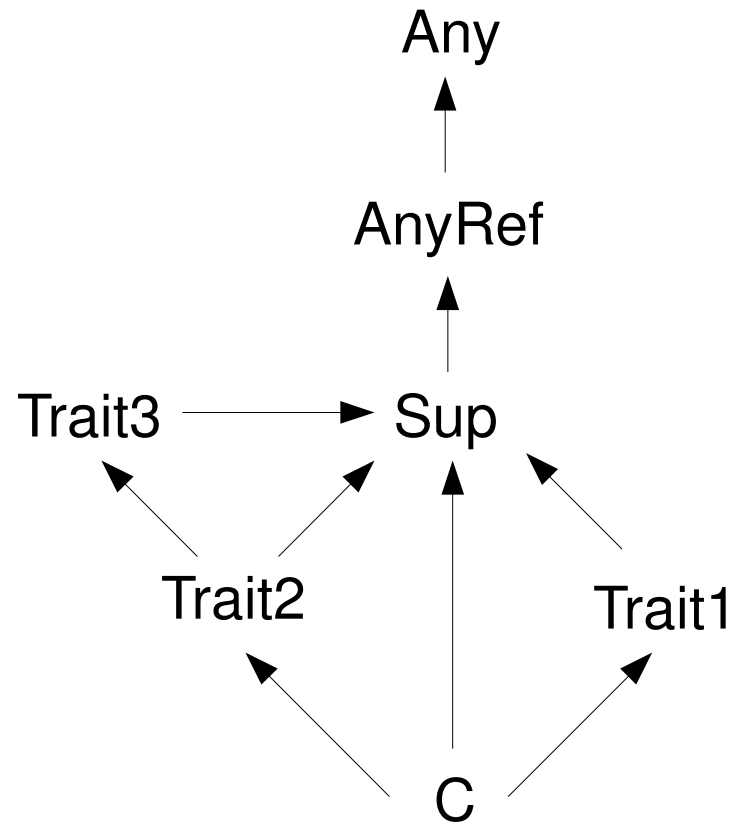
The linearization of C , $L(C)$ is defined as follows:

$$L(C) = C, L(C_n) \overset{\blacktriangleright}{+} \dots \overset{\blacktriangleright}{+} L(C_1)$$

Here $\overset{\blacktriangleright}{+}$ denotes concatenation where elements of the right operand replace identical elements of the left operand:

$$\begin{aligned} \{a, A\} \overset{\blacktriangleright}{+} B &= a, (A \overset{\blacktriangleright}{+} B) \text{ if } a \in B \\ &= A \overset{\blacktriangleright}{+} B \quad \text{if } a \notin B \end{aligned}$$

Example



```
class C extends Sup with Trait1 with Trait2
```

Example

Class	Linearization
Sup	Sup, AnyRef, Any
Trait1	
Trait2	
C	..., Sup, AnyRef, Any

Example

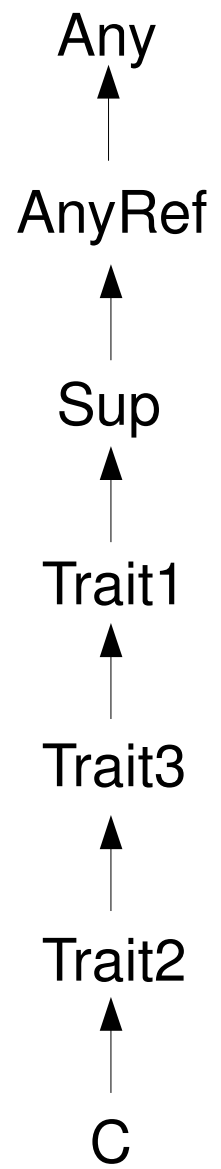
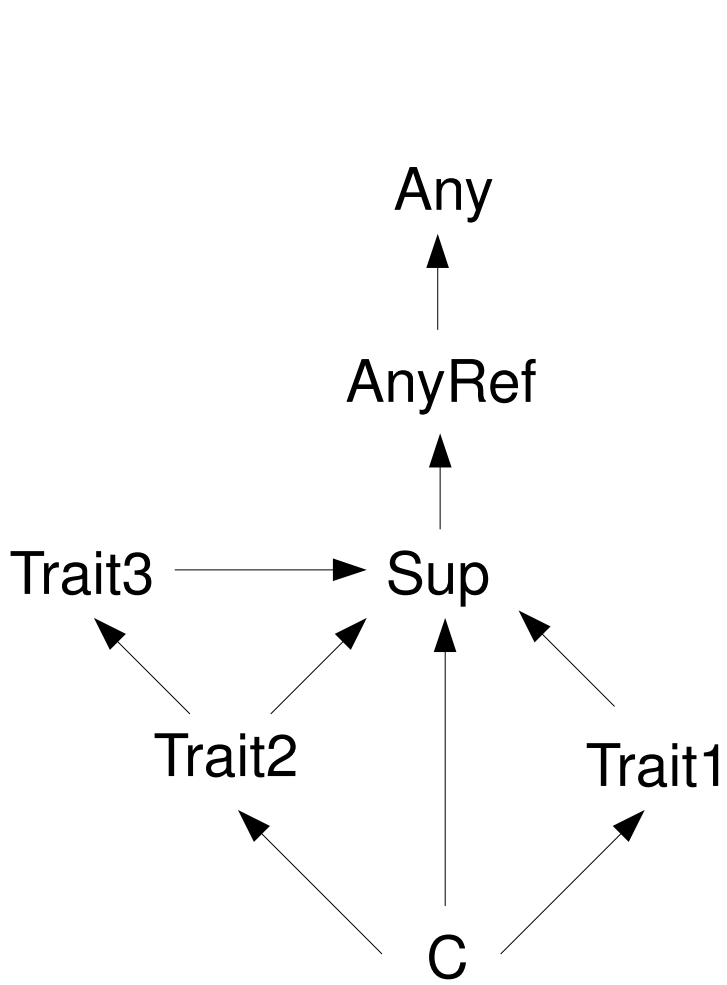
Class	Linearization
Sup	Sup, AnyRef, Any
Trait1	Trait1, Sup, AnyRef, Any
Trait2	
C	..., Trait1, Sup, AnyRef, Any

Example

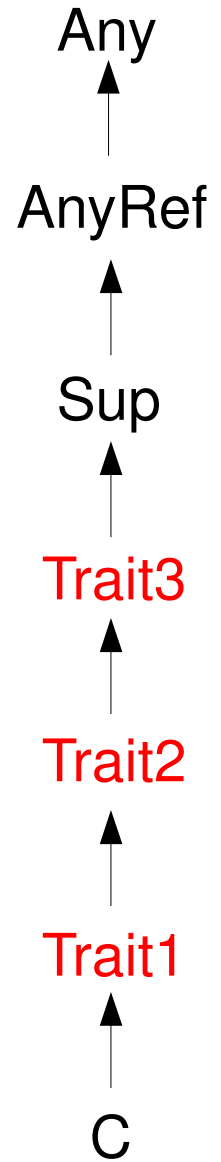
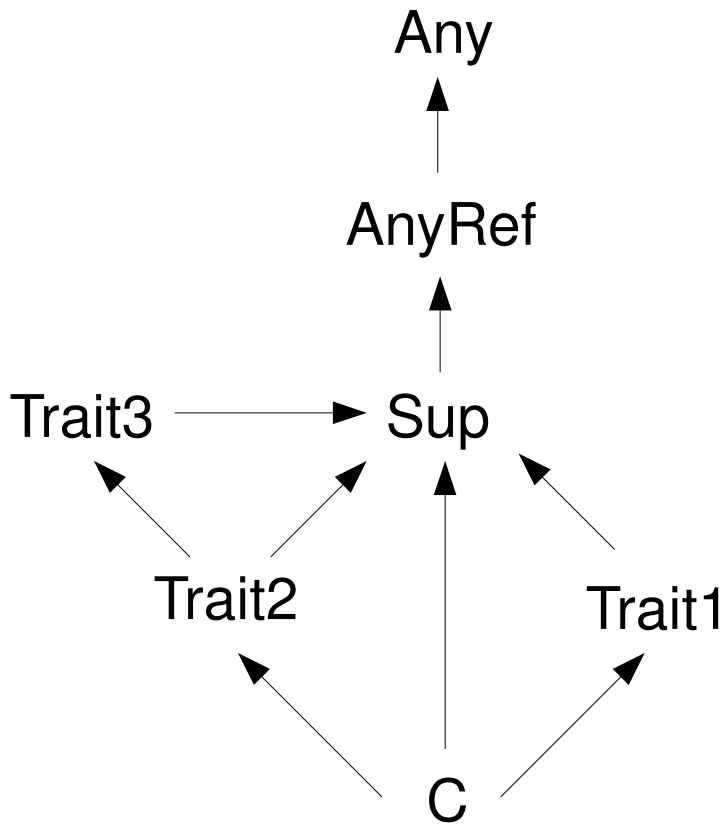
Class	Linearization
Sup	Sup, AnyRef, Any
Trait1	Trait1, Sup, AnyRef, Any
Trait2	Trait2, Trait3, Sup, AnyRef, Any
C	..., Trait2, Trait3, Trait1, Sup, AnyRef, Any

Example

Class	Linearization
Sup	Sup, AnyRef, Any
Trait1	Trait1, Sup, AnyRef, Any
Trait2	Trait2, Trait3, Sup, AnyRef, Any
C	C, Trait2, Trait3, Trait1, Sup, AnyRef, Any



```
class C extends Sup with Trait1 with Trait2
```



`class C extends Sup with Trait2 with Trait1`