Figure 1: Behavioral preserving transformation applied by Eclipse. JRRT rejects applying it and warns a message: cannot access method m.

Listing 1: Before Refactoring

```
package p1;
import p0.*;
public class A {
  public B f = null;
  protected long m(long a) {
    return 1;
  }
  public long m(int a) {
    return 0;
  }
}

package p0;
import p1.*;
public class B extends A {

}

package p1;
import p0.*;
public class C extends B {
  public long test() {
    return super.m(2);
  }
}
```

Listing 2: After Refactoring

```
package p1;
import p0.*;
public class A {
  public B f = null;
  protected long m(long a) {
    return 1;
  }
}

package p0;
import p1.*;
public class B extends A {
  public long m(int a) {
    return 0;
  }
}

package p1;
import p0.*;
public class C extends B {
  public long test() {
    return super.m(2);
  }
}
```