

Subject: Resolve generic without invoking a procedure or evaluating arguments
From: Van Snyder
Reference: 04-273

1 **Number**

2 TBD

3 **Title**

4 Resolve generic without invoking a procedure or evaluating arguments

5 **Submitted By**

6 J3

7 **Status**

8 For consideration.

9 **Basic Functionality**

10 Given exemplars of actual arguments, resolve a generic name to a specific procedure without invoking
11 the procedure or evaluating its arguments.

12 **Rationale**

13 With care and diligence, one can develop a program so that related sets of variables, constants and
14 function results are parameterized by a single kind type parameter. In order to change the kind of that
15 set of entities, one need only change one named constant’s definition — almost: Generic procedures
16 cannot be actual arguments or procedure pointer targets. Thus, if one needs to change the program, in
17 addition to changing the single named constant definition, one needs to find all places where a specific
18 procedure that operates on the entities in question is an actual argument or procedure pointer target,
19 and manually edit those appearances.

20 It would be helpful to have a facility to resolve a generic name to a specific procedure without evaluating
21 any arguments or invoking a procedure.

22 **Estimated Impact**

23 Minor. Processors already know how to do generic resolution. Estimated at meeting 169 to be at 4 on
24 the JKR scale.

25 **Detailed Specification**

26 Given exemplars of actual arguments, resolve a generic name to a specific procedure without invoking
27 the procedure or evaluating its arguments.

28 There are at least two ways to do this. One is to provide a syntax that is suggestive of procedure
29 reference, but does resolution instead. One possibility for this is to enclose an actual argument list in
30 square brackets or curly brackets instead of round brackets. E.g.,

```
31 call solver ( myVec, myJacobian, myModel[myVec,myJacobian] )
```

32 Another is to provide an entity that looks like an intrinsic function but that has the important distinction
33 that its arguments aren’t evaluated. Indeed, this entity that has the appearance of a function reference
34 isn’t even invoked during program execution. It is entirely resolved to a procedure by the processor
35 during translation. E.g.,

1 call solver (myVec, myJacobian, resolve(myModel,myVec,myJacobian))

2 RESOLVE would be a an inquiry “function” that takes a *generic-spec* as its first “argument,” and
3 doesn’t evaluate its other arguments — because it’s an inquiry function.

4 It should be possible to resolve a type-bound generic reference, e.g., `resolve(a%b)`.

5 No matter what syntax is used, it should be allowed to use the result either as an actual argument or a
6 procedure pointer target.