

# Dynamic binding

## Contents

- [Overview](#)
- [Object variable in parameter list](#)
- [Use classlocal variables with object variable](#)
- [Call class method for object variable](#)
- [Access class property for object variable](#)

## Overview

The HBasic compiler normally tries to identify a class member exactly and creates code to access this class member. If for example a method within a class should be called the parser compares the parameter types of the method call with the parameter types of the method definition and creates a call to the method implementation with matching parameters. In combination with variables of type object it may not be possible for the compiler at compiletime to decide which is the correct method implementation. In this case the parser will create a dynamic method call and the runtime environment must decide which is the correct method body for the actual parameter types.

```
Sub foo( i As Integer )
...
End Sub

Sub foo( d As Double )
...
End Sub

Dim o As Object

Sub button1_clicked()
    o = 1
    foo(o)
    o = 22.33
    foo(o)
End Sub
```

**Example dyn\_call\_method3.bas:** Example program for dynamic binding.

Currently the HBasic parser creates dynamic access in three situations:

1. If a method is overloaded and a variable of type object will be used in the parameter list (see example above)
2. If a member of a class will be referenced and the class reference is stored in a variable of type object
3. If a method of a class allows overloading in an inherited class definition

The example shows some code where an object variable holds the reference to an instance of a user defined class and a class member will be used. You may find some working examples in this document.

```
Class c
    << class member definitions >>
End Class

Dim o As Object

Sub button1_clicked()
    o = New c()
    o.member()
End Sub
```

**Example :** Calling a class method for an object referenced in a variable of type Object

The following example shows some code where a method foo may be overloaded in the inherited class definition c2. This also needs dynamic binding and will be implemented in HBasic next.

```
Class c1
    Sub foo( i As integer )
    ...
End Sub

Class c2 Inherits c1
    Sub foo( d As Double )
    ...
End Sub
```

**Example :** Example that needs dynamic binding because of inheritance.

The rest of this document shows some example programs where dynamic binding must be used by the HBasic parser.

**Implementation details:** To handle dynamic binding HBasic must decide at runtime, which of the two sub bodies should be called. This is only possible if HBasic can access the signature of every subroutine at runtime. This includes a description of the parameters and their types and which address should be called with a selected parameter list.

The interpreter can use the sub and class description created by the parser but the compiler must create a special description of the subroutine parameter lists in the executable code to handle this. If the access to the member will be made through a variable of type object *o.member* the parser searches for a parameter list following the runtime environment and generates either a method call or a property access with unknown base class.

The runtime environment must have a description of the classlocal variables, properties and methods for each class. For each property the name and type will be stored and for each method the name and the parameter list will be stored.

Special runtime function compare the types of the actual parameters with the signature of the subroutine description and call a matching subroutine code if it can find one. Otherwise a runtime error will be triggered. This functions must also be capable of handling inheritance for classes.

## Object variable in parameter list

```
Sub a( i As Integer )
    Print "a called with integer parameter"
End Sub

Sub a( d As Double )
    Print "a called with Double parameter"
End Sub

Dim o As Object

Sub button1_clicked()
    o = 1
    a(o)
    o = 22.33
    a(o)
End Sub
```

**Example dyn\_call\_method3.bas:** Example program for dynamic binding.

## Using classlocal variables

Example 2: Access classlocal variable for class in object variable

If classlocal variables will be used from within the class body the runtime code has to get the pointer to the class memory in another way. Since this sourcecode examples will also be used to test HBasic functionality the method inclass will be called to check access to i from within the class.

```
Class c
    Dim i As Integer

    Sub inclass( newval As Integer )
        i = newval
        Print i
    End Sub
End Class

Dim o As Object

Sub button1_clicked()

    o = New c()
    o.i = 1122
    Print o.i

    o.inclass( 2233 )
End Sub
```

**Example :** Example where dynamic binding will be used to access classlocal variables

## Call method for class in object variable

Example 3: Call method for class in object variable

```

Class c

    Sub a( i As Integer )
        Print "Method a with integer parameter"
        Print i
    End Sub

    Sub b( i As Integer )
        Print "Method b with integer parameter"
        Print i
    End Sub

    Sub b( d As Double )
        Print "Method b with double parameter"
    End Sub

End Class

Dim o As Object

Sub button1_clicked()
    o = New c()

    o.b( 11 )
    o.b( 22.33 )
End Sub

```

**Example :** Example where dynamic binding will be used to access a class method

## Read and write property from class

Example 4: Use property for class in object variable

```

Class c
    Dim i As Integer

    Property val As Integer

    Get
        Return i
    End Sub

    Set
        i = value
    End Set

End Class

Dim o As Object

Sub button1_clicked()
    o = New c()

    o.val = 2233
    Print o.val
End Sub

```

**Example :** Example where dynamic binding will be used to access a class property