SUMMARY
This Tech Note outlines the benefits, examples, and limitations of using indirect references in Industrial Application Server.

Benefits of Using Indirect Variables

Pre-2.0
With previous versions of IAS, the only way to reference one attribute to another was through InputOutput Extensions. In the following example, Object_01.UDA_01 is assigned to Object_FacePlate.UDA_01. After this is script is executed, any changes made to Object_FacePlate.UDA_01 will be sent to Object_01.UDA_01 and vice versa.

Object_FacePlate.UDA_01.InputSource = "Object_01.UDA_01";

The problem with this is that it may take one complete scan to pass a value from one to another. In the above example, if a value is written to Object_FacePlate.UDA_01, that value will not be written to Object_01.UDA_01 until the next scan.

What 2.0 Brings
With Industrial Application Server 2.0, indirect references can be made by binding a local indirect variable to an attribute. Once this binding has been completed, values can be read from and written to the referenced attributes instantly.

DIM ind_Value AS Indirect;

ind_Value.BindTo("Object_01.UDA_01");

Examples

Example 1: Toggling a Boolean UDA
In the following example, a local indirect variable is defined, bound to a previously defined user-defined attribute (UDA_01), then toggled. Once this script has run, it will toggle UDA_01. The script and the UDA are defined in the same UserDefined object.

DIM ind_Value AS Indirect;

ind_Value.BindTo("me.UDA_01");
ind_Value = NOT ind_Value;

Example 2: Resetting an Array
The following example illustrates how to quickly clear the values of an array. Once this script has run, it will set all the values of the 10-dimensional array, UDA_Array, to zero. The script and the UDA are defined in the same UserDefined object.

DIM ind_Value AS Indirect;
DIM inc AS Integer;

FOR inc = 1 to 10
    ind_Value.BindTo("me.UDA_Array[" + Text(inc, ") + "]");
    ind_Value = 0;
NEXT;

Limitations
The following limitations apply to indirect variables:

- They can only be defined as local variables.
- They cannot be used to bind attributes from objects running on separate engines.