Subject: Use Identifiers or Keywords
Posted by Robbie on Thu, 05 Oct 2006 04:42:55 GMT
View Forum Message <> Reply to Message

--- Warning to all ITTVIS employees ---
--- This post contains product names which might offend some viewers.

Here is another little problem which I'm trying to solve using IDL.
It's a very specific problem, so I won't be offended by a sea of blank looks.

I have a framework which is a very partial implementation of a dumped RSI product (Watsyn). This framework has a workflow engine which executes several programs in order.
Each program might receive input from a previous program. If any particular program is run, then the changes must be propogated through the system. For the record - I'm currently using IDLitMessage to send the messages about these changes.

I connect programs (objects) together, but the syntax might go something like this:

```
obj_a -> RegisterOutput, 'OUTPUT1'
obj_b -> RegisterInput, 'INPUT1'
result = obj_b -> Connect(obj_a,'INPUT1','OUTPUT1')
```

In plain english this just means:

Object A has an OUTPUT
Object B has an INPUT
Connect the OUTPUT of object A to the INPUT of object B

The problem I have is that it is very easy for the end user to misspell, capitalise or add whitespace. This is a particular problem if there are many connections. One possible way around this is to use keywords.

```
obj_a -> RegisterOutput, /OUTPUT1
obj_b -> RegisterInput, /INPUT1
result = obj_b -> Connect(obj_a -> GetConnector(/INPUT1),/OUTPUT1)
```

This would make it possible to specify the accepted keywords at compile time, and also limit ambiguity.

Another syntax might be:

```
obj_a -> RegisterOutput, OUTPUT1=obj_o
```

```
obj_b -> RegisterInput, INPUT1=obj_i
result = obj_o -> Connect(obj_i)
```

This syntax is not so lightweight because it involves initialising two objects for each connection.

Any thoughts?

Robbie