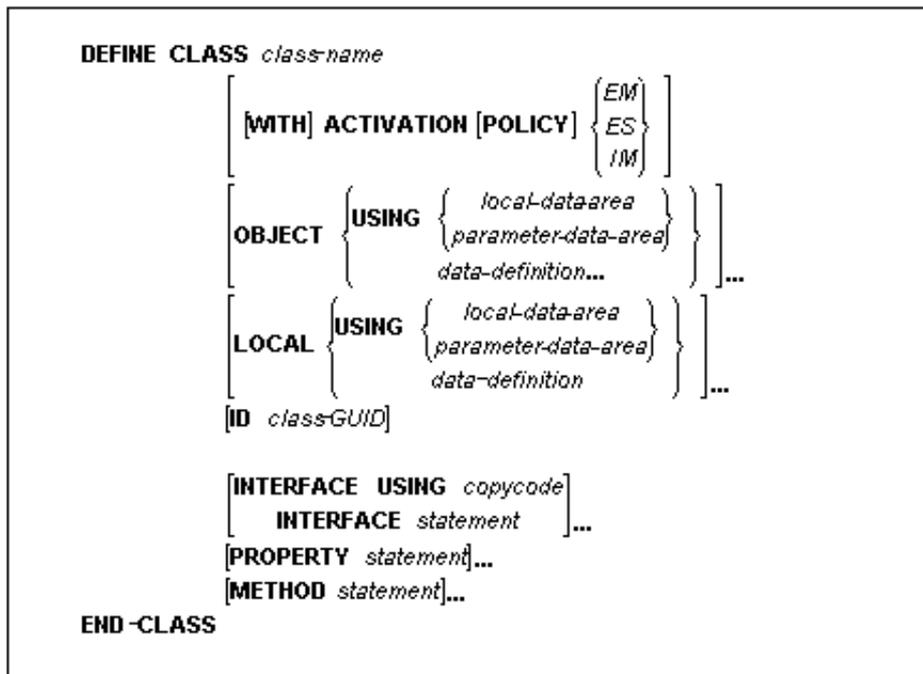


DEFINE CLASS



Function

The DEFINE CLASS statement is used to specify a class from within a Natural class module.

A Natural class module consists of one DEFINE CLASS statement followed by an END statement.

class-name

This is the name that is used by clients to create objects of this class. The name can be up to a maximum of 32 characters long. The name may contain periods: this can be used to construct class names such as <company-name>.<application-name>.<class-name>. Each part between the periods (...) must conform to the Natural naming conventions for user variables (please refer to the Natural Reference documentation for further information).

If the class is planned to be used by clients written in different programming languages, the class name should be chosen in a way that it does not conflict with the naming conventions that apply in these languages. Bolero for example uses the Java naming convention. So, a class that is planned to be used in a Bolero client should also follow the Java naming conventions.

WITH ACTIVATION POLICY Clause

The WITH ACTIVATION POLICY clause is used to define explicitly the activation policy which is registered for the current class.

You can set the following parameters:

Parameter	Description
EM	Sets activation policy <i>ExternalMultiple</i>
ES	Sets activation policy <i>ExternalSingle</i>
IM	Sets activation policy <i>InternalMultiple</i>

When the class is stored and registered, the setting in the WITH ACTIVATION POLICY clause overrides the ACTPOLICY=*activation-policy* (for OS/390, DCOM=(ACTPOL=*activation-policy*)) profile parameter, but is in turn overridden by manual registration using the REGISTER command with an explicit activation policy definition. For further information, see the section Activation Policies.

OBJECT Clause

The OBJECT clause is used to define the object data. The syntax of the OBJECT clause is the same as for the LOCAL clause of the DEFINE DATA statement. For further information, see the description of the LOCAL clause of the DEFINE DATA statement.

LOCAL Clause

The LOCAL clause is only used to include globally unique IDs (GUIDs) in the class definition. GUIDs need only be defined if a class is to be registered with DCOM. GUIDs are mostly defined in a local data area. For further information, see the section Globally Unique Identifiers (GUIDs).

The syntax of the LOCAL clause is the same as for the LOCAL clause of the DEFINE DATA statement. For further information, see the description of the LOCAL clause of the DEFINE DATA statement.

ID Clause

The ID clause is used to assign a globally unique ID for the class. The class GUID is the name of a GUID defined in the data area that is included by the LOCAL clause. The class GUID is a (named) alphanumeric constant. A GUID must be assigned to a class if it is to be registered with DCOM.

INTERFACE USING Clause

The INTERFACE USING clause is used to include copycode that contains INTERFACE statements.

copycode

The copycode used by the INTERFACE USING clause may contain one or more INTERFACE statements.

See the following statements for further information:

- INTERFACE
- PROPERTY
- METHOD