Object Reference

All functions in the network IVR are carried out using objects. An object of a particular type is used to execute particular functions. This section discusses all available objects in alphabetical order.

Every object has a name, an input and (optionally) one or more outputs. Most objects have one or more outputs. Example:

Every instance of an object needs to be parameterised to define its behaviour.

Example: For example, an object of type *Email* accesses the functionality to send an email. An instance of this object is parameterised, to send the email to a particular address with a particular subject and content.

You have already learned how to use the graphical editor whilst working through the Programming Example.

Object Name

To identify an object uniquely in the application, it requires a name. The internal ID of the object (a number in brackets) is automatically created when the object is created. This can be changed or removed.

Input

Most objects have one or more outputs.

Outputs

Most objects have one or more outputs. The outputs represent different events and states that can occur when the object is executed.

Connected Outputs

If an output is connected with a further object, the program execution continues with the connected object if the corresponding event occurs during object execution.

Not Connected Outputs

If an output is not connected, then the behaviour is different depending on the object. In most cases, if an output is not connected, then the routing application ends: Program execution will be stopped, the telephony connection will be hung up. This does not apply to all objects. In menu objects a not connected output for a DTMF key signifies that this option is not active. Program execution continues within the object.

Object Reference

The following section describes all objects in alphabetical order.