BPA
Platform

Technical Overview
Call COM Object Tool

codeless
PLATFORMS

Copyright

Head Office:

Codeless Platforms

Suite 1 & 2 Bourne Gate

25 Bourne Valley Road

Poole

BH12 1DY

United Kingdom

Tel: +44 (0) 330 99 88 700

Email: enquiries@codelessplatforms.com

Trademarks

Contents

Introduction

The **Call COM Object** tool is used to create a task step that calls third party COM Objects to provide integration with other systems, protocols, or applications. The COM Object may be provided by a COM Server (`.EXE`) or COM `.DLL`, written in languages such as Microsoft Visual Basic or C++, or others supporting COM and DCOM. Using this step, BPA Platform can call an object and pass either static or dynamic parameters to the object in the method call.

Features

▶ COM Server (`.EXE`) or COM `.DLL`

▶ Pass static and/or dynamic data from other steps

▶ Pass Recordset and or Document data

▶ Call failure options

▶ Memory features to memorise task data created at run-time

Working with Other Tools

The other tools that the **Call COM Object** step interacts directly with, whether by exposing information to it or consuming information from it, are listed below.

Consuming from Other Tools

Icon	Tool Name	Tool Category
	Database Query (ODBC)	Input and Data Connectors
	Database Query (OLEDB)	Input and Data Connectors
	Import Flat File	Input
	Convert XML to Recordset	Format
	Create Workflow Job	Format
	Format as Flat File	Format
	Format as HTML	Format
	Format as HTML Pro	Format
	Format as Text	Format
	Run Microsoft Reporting Services	Format
	Run Microsoft Word (Merge)	Format
	Transform Data	Format
	Filter Data	General

Objects Consumed

Objects that the **Call COM Object** Step consumes that are exposed by other Steps are listed below.

- ▶ **Recordset** — Information structured into columns and rows

- ▶ **Documents (Any)** — Documents of any type

- ▶ **Variables (Text)** — Textual data is extracted from selected Variables

Exposing to Other Tools

The **Call COM Object** tool does not directly interact with other tools through exposure. However, it does expose objects that can be used by other tools — see Objects Exposed.

Objects Exposed

Objects that the **Call COM Object** Step exposes that can be utilised by other Steps are listed below.

- ▶ **RecordSource** — If an **Input Recordset** has been selected from a Step previously created in the Task, then it displays the columns included in the recordset.

- ▶ **DocumentSource** — If a **Document Source** has been selected from a Step previously created in the Task, then it displays the columns included in the recordset used by the Step.

- ▶ **Memory Definitions** — Displays a list of **Memory Definitions** created for the Step and the columns to be memorised for each one. Double-clicking on a Memory Definition then opens the applicable memory configuration dialog box to enable the details to be edited.

- ▶ **Step Properties** — A number of properties for the Step are exposed in the Environment tab of the Task Browser dialog box

Global Configuration

Global configuration is not required for the **Call COM Object** tool.

Step Configuration

The **Call COM Object** tool does not require a global connection to a COM object before being added to a task.

To add a new **Call COM Object** step to an existing task, you either:

▶ Click and drag the **Call COM Object** icon from the **Task Browser** to the task **Design** area.

▶ From the task's **Design** tab, right-click on empty space and select *Add > Execute > Call COM Object.*

For a detailed description of how to create new tasks, refer to the product help.

About the General Tab
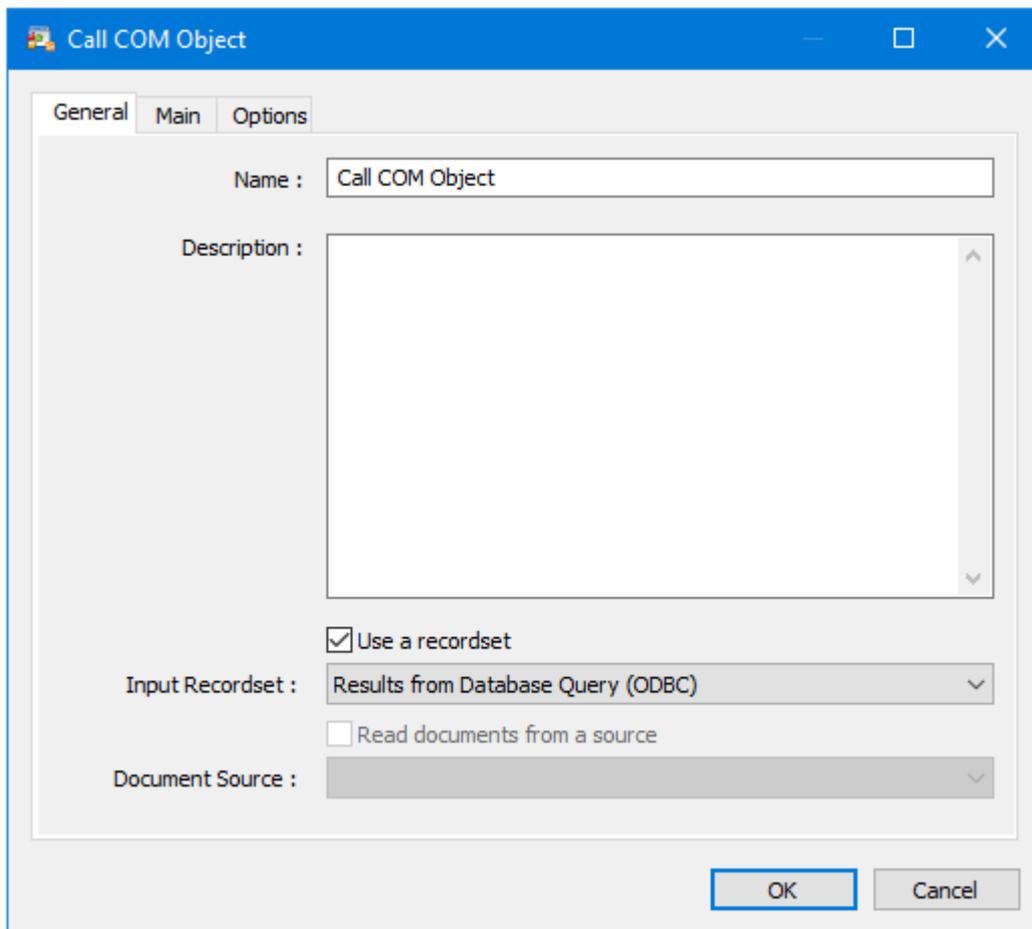
The **General** tab is used to enter the following details for the **Call COM Object** step:

▶ **Name** — A unique name for the step may be manually entered. The name is used to identify the step within the task and particularly within the task plan. It is advisable to name the steps as accurately as possible so that the task plan itself gives the user a good impression of the overall task process at a glance.

▶ **Description** — A description for the step may be manually entered. The description is used to explain what the step does and how it is configured to achieve this. When maintaining the task, this helps users understand how the task works.

▶ **Use a recordset** — When selected the **Input recordset** drop down list is active to enable the output recordset from another task step to be used as the step input.

☐ **Input a Recordset** — All available recordsets from **Input** steps previously created in the task are displayed in a drop down list for selection

▶ **Read documents from a source** — When selected the **Document Source** drop down list is active to enable the output from another task step to be used as the step input.

☐ **Document Source** — All available documents from **Format** steps previously created in the task are displayed in a drop-down list for selection. Note that if a Crystal Report is being used as the document source, then a number of data formats are displayed for selection.

About the Main Tab
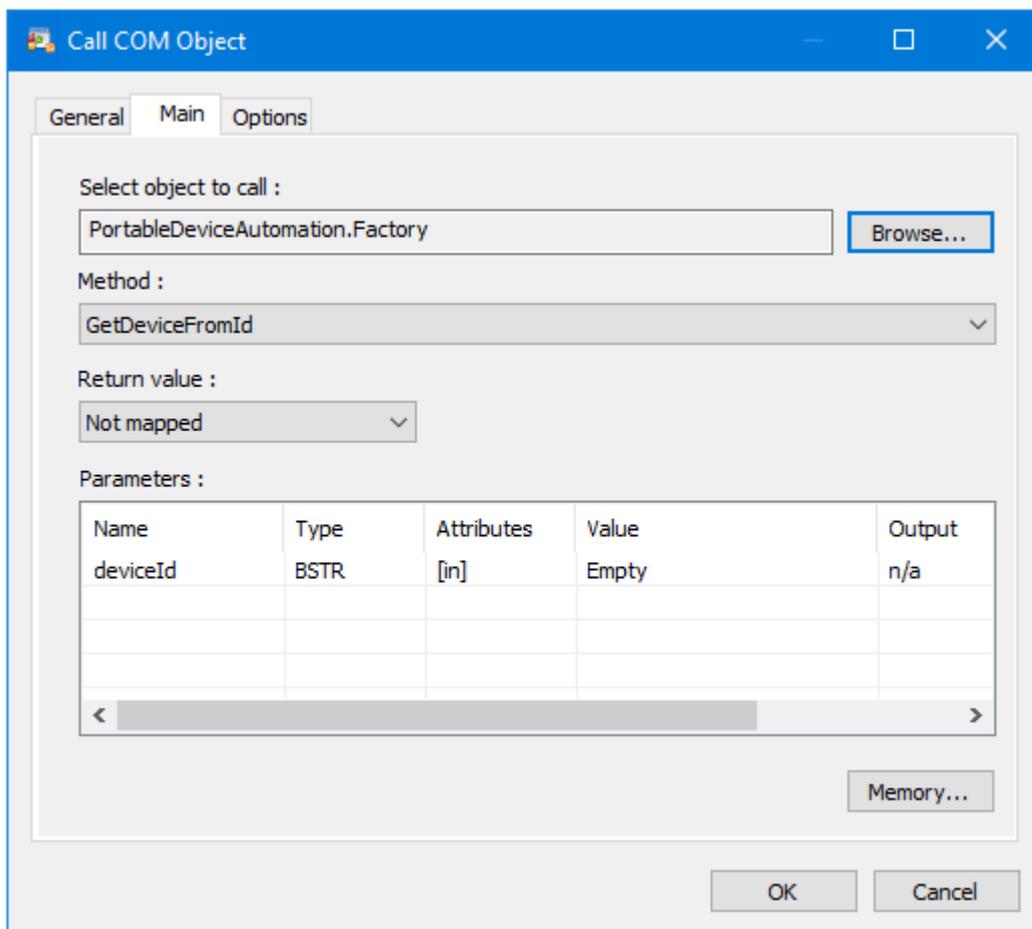
The **Main** tab is used to select a COM object method and set values for its parameters.

The tab contains the items listed below.

▶ **Select object to call** — An object method may be selected through the **Browse** button which pops up the Object Browser dialog box. The selected object is then displayed.
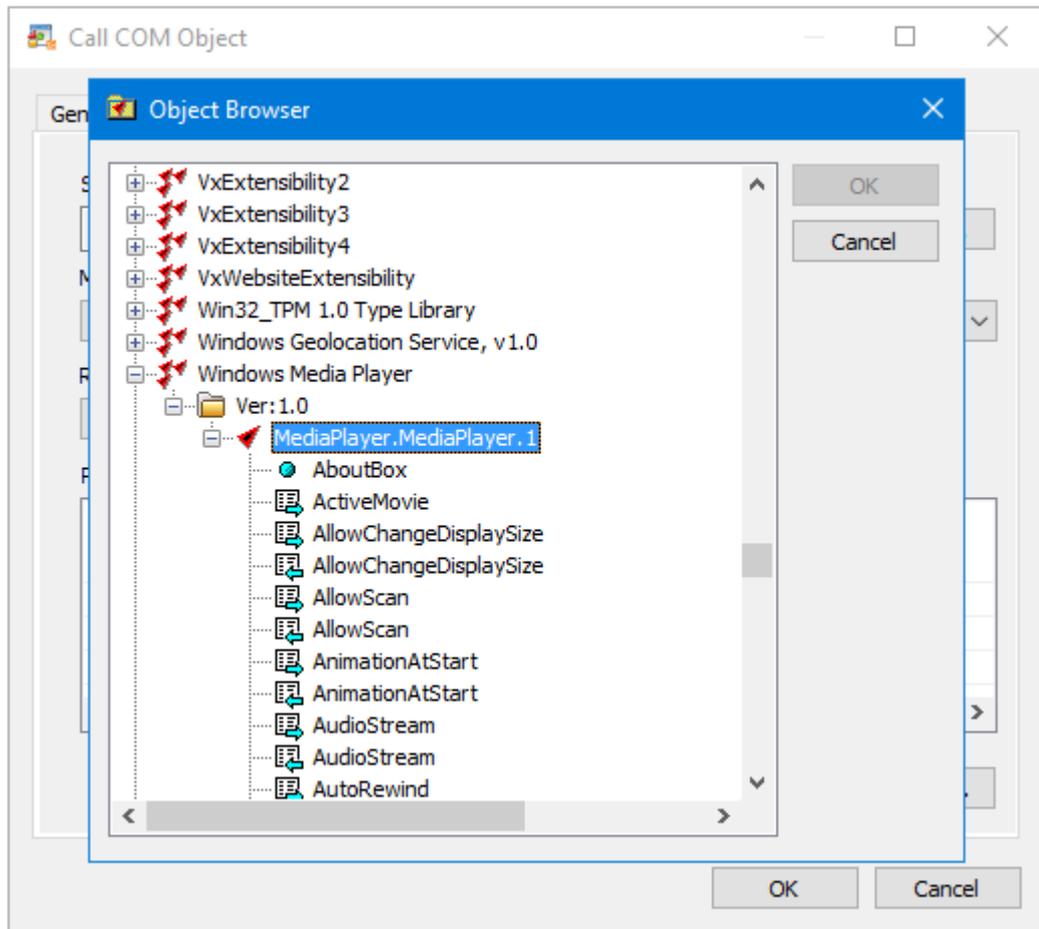
▶ **Method** — The method selected in the browser screen above is displayed. The drop down list also displays the other methods available for the selected object.

▶ **Return Value** — Displays a list of variables for selection to which the return value may be mapped. Only one variable may be selected.

▶ **Parameters** — Displays a list of parameter properties for the selected method under the following headings:

☐ **Name** — Displays the parameter name

☐ **Type** — Displays the data type.

☐ **Attributes** — Displays the attributes.

☐ **Value** — Displays the default value which may be manually edited or a column, property or variable containing the required parameter value dragged and dropped from the Task Browser dialog box to enter a value dynamically.

☐ **Output** — Displays a list of variables for selection. Where a COM Object has one or more "output" parameters, they can be mapped to a Task variable.

▶ **Memory** — Displays the Memory Definitions dialog box through which an Internal, ODBC, OLEDB or **Call Stored Procedure (OLEDB)** memory may be created to memorise some or all of the recordset used by the Step.

About the Options Tab

The **Options** tab allows you to define how errors in this step are handled at task runtime.
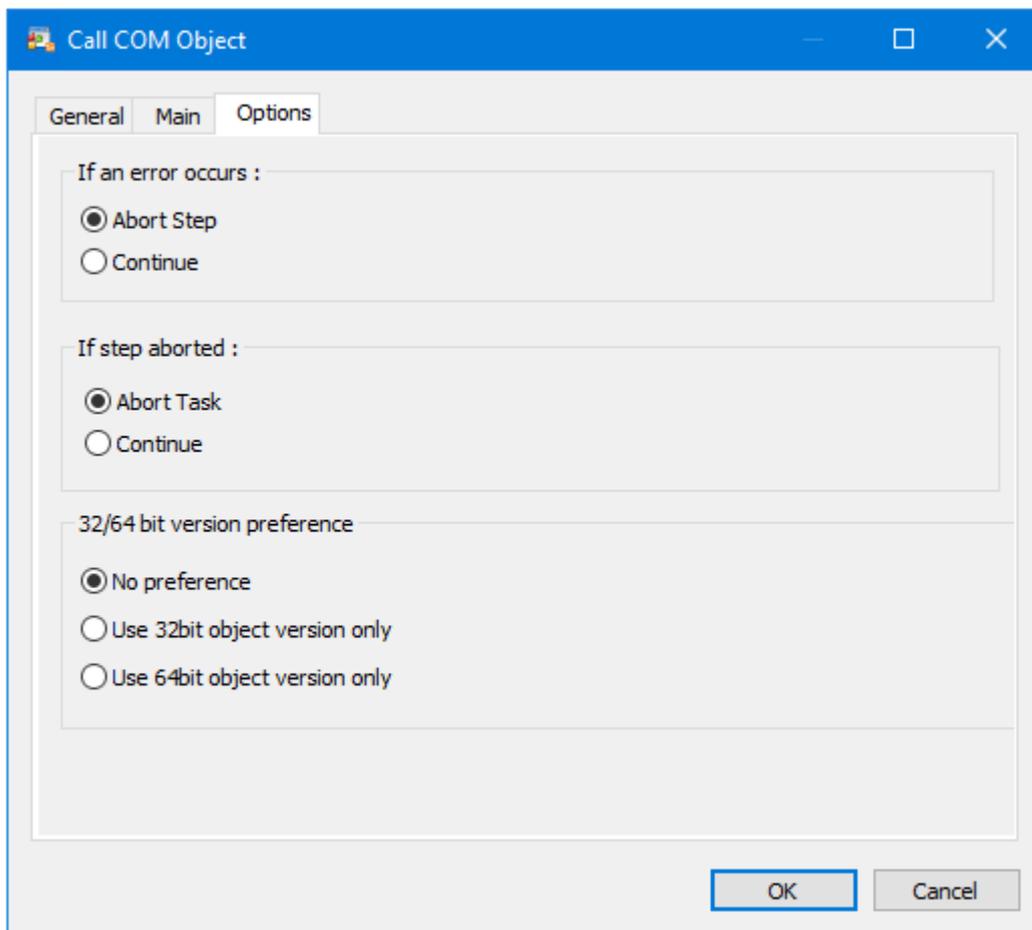
If an error occurs, you can decide whether the step should **Continue** processing, or terminate the step immediately (**Abort Step**).

If the step is aborted, you can choose to **Continue** processing onto the next step in the task, or terminate the whole task immediately (**Abort Task**).

Select whether 32- or 64-bit COM object is perferred.

Want to learn more?

Discover how Codeless Platforms can help your business by improving performance, boosting efficiency and cutting costs.

📱 +44 (0) 330 99 88 700

✉ enquiries@codelessplatforms.com

🧭 www.codelessplatforms.com

codeless PLATFORMS