

opus

Pro Features

Digital Workshop

Copyright © 1997– 2005 Digital Workshop – All rights reserved

Contents

Introduction	5
Welcome	7
Overview	9
Databases	11
Databases - Overview	13
Getting Started	13
Showing Records From a Database	15
Making databases interactive	17
Connect to a Database	19
Data Source Names (DSN's)	19
Creating a new DSN	20
Re-using DSNs	21
Creating a Database Query	23
The SQL Wizard	23
Using the LIKE Operator	27
Matching Dates and Times	28
Actions: Database	29
Database Query	29
Next Record	31
Previous Record	31
First Record	32
Last Record	32
Delete Record	32
Update Current Record	33
Insert New Record	34
End Query	36
Number of Records	36
Current Record Number	38
The Resource Manager	41
The Resource Manager	43
Resource Manager Workspace	43
The Toolbars	45
Filter Resources	47
Set\Clear Resources	49
Consolidate Resources	51
Change Data Paths	54
Purge Resources	55
Additional Features	59
Arrays	61
Creating Arrays	61
ListBoxes and Tables	65
ListBoxes	65
Creating a ListBox	65
Create As Table	67
Layout	71
Panel Properties	72

Setting Panels to Chapters.....	73
Publication Properties	75
Additional Resources	75
Removing Resources.....	77
Editing Resources	77
Database.....	78
Data Sources	79
Editor.....	79
'Don't Update Data Source' Check Box.....	79
Published	80
File Location	81
Image Background.....	81
Minimise Options.....	81
Custom Triggers	83
Activating a Custom Trigger using OpusScript	84
Using Parameters	85
How to use Parameters.....	85
Additional Launch Actions	89
Post Web Data	89
Clock Actions	93
Start Clock	93
Stop Clock	94
Pause Clock.....	95
Continue Clock	95
File Management Actions	97
File Browse.....	97
Copy File.....	99
Delete File	100
Conditional Actions	103
Select	103
Case.....	105
Information Tab	107
Permissions	109
Overview.....	109
Password Protection.....	110
Using the Permission Settings.....	110
Lock Out	110
Custom Settings	111
Index	113

Introduction

Welcome...7

Overview...9

Welcome

Welcome to Opus, a range of multimedia authoring and creativity tools for quickly and easily creating all types of interactive publications. This manual details the major elements of the professional version which extends the ease of use of the standard versions of Opus by including database connectivity via the industry-standard ODBC functionality. It also adds a versatile scripting language, which is based on the syntax of the widespread Javascript language.

Together with additional actions and objects, the Professional versions of the Opus range provide developers of all levels with a rapid multimedia development tool which is both easy to get started with and powerful enough for a wide variety of tasks.

Developing spectacular, modern, media-rich programs and web sites has surely never been easier.

Innovate - Create - Communicate - Beyond Extraordinary...

Overview

The basic functions of Opus are described in the *User Guide* manual - if you are new to Opus or to multimedia authoring, we recommend that you read this manual first, to familiarise yourself with the Opus workspace and the wide range of tools and functions available to you.

This manual covers the features contained within the Pro versions of Opus, including database connectivity and advanced actions and objects. The Pro versions also include OpusScript which is detailed in a separate reference manual and, of course, in the Help file.

The manual is split into the following sections:

Databases Section

The first section looks at Databases and describes how to: connect to a database; build a query to select records from a database; and interact with a database using special Database Actions. In addition, the *OpusScript Functions* section has a full list of functions that have been exclusively designed for working with Databases. In addition, the 'Publication Properties' chapter in the *Additional Features* section, describes the **Database** tab in the Publication Properties dialog box.

The Resource Manager Section

The second section describes the Resource Manager, an incredibly powerful tool that allows you to review, update, move and delete resources in a publication. Managing your resources when developing your publication is very important and this tool provides a one-stop shop for all your resourcing needs.

Additional Features Section

The final section of this manual describes all the new features that are available in the professional version of Opus. There are many new Actions and Triggers that you can explore. Some of the Actions and Triggers have been added to existing Action and Trigger tabs in the Actions dialog. Other Actions and Triggers are in new tabs that have been added alongside the existing tabs in the Actions dialog. This

manual will only cover the additional features, all other Actions and Triggers are contained in the '*User Guide*'.

Databases

Overview...	13
Showing Record from a Database...	15
Connect to a Database...	19
Creating a Database Query...	23
Actions: Database...	29

Databases - Overview

Databases are useful for storing a variety of information, such as, Employee details and Catalogues of company products. However, databases are not very exciting to look at. Professional versions let you connect to databases and to show records on pages within your publication, users can then interact with databases in a more stimulating and creative environment. For developers, the new database feature allows them not only to access information but to write information to databases within an Opus publication, such as, Login ID's and Passwords or the results of tests taken within a publication.

Getting Started

All of the Database features are provided within the Actions dialog under a new tab named 'Database'. There are a variety of actions available, such as, getting the next record, updating a record and inserting new records. However, before you can do any of these actions you must first do the following:

1: Connect to a database (see *Connect to a Database* section for more information).

2: Create a Query that will select records from a database (see *Create a Database Query* section for more information).

If you would like a brief guide on using a database in your publication - see *Showing Records From a Database* section for more information.

For information on setting passwords and protecting access to databases within an Opus publication, a new tab has been added to the Properties dialog named 'Database' (see *Publication Properties: Database* for more information).

Showing Records From a Database

Databases are often used to store information, such as product information, historical facts, travel information, price information, employee information, and much more. With Opus you can display this information in a more attractive and stimulating environment as well as allow users to navigate around the database.

To show records from a database:

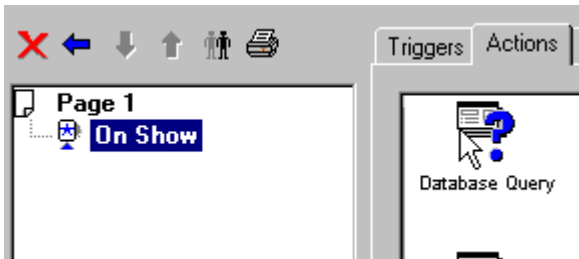
1: First you need to decide how the information will be displayed. Will the first record appear as they enter the page or do they press a button to show the records?

If you want to have the first record appear on the page as soon as the page is displayed, then open the **Actions** dialog for the page and select the **On Show** trigger.

2: Once you have selected the trigger that will start the process, select the Database tab from the list of mini tabs at the bottom of the Actions tab.



3: Double-click on the **Database Query** icon to add it to the On Show trigger. A Database Query tab will appear after the Programming tab in the Actions dialog, this is used to connect to the database and set up the query list, that is, the records you want to select from the database. (see Database Query in *Actions: Database* for more information).



The Database Query action also performs one more vital role, it is used to match a field from your database to a variable name in your publication. What this means is that the contents of the field of the current record will be added to the contents of the variable you asked it to save it in, as in this example:

Field	Variable
Prod_id	product_id
Prod_name	ProductName
Short_desc	ShortDescription

4: When you have filled in the Database Query information you will need to create text fields on the page in which the data from the first record in the database will be displayed, like this...



5: Use the **Insert Variable** tool to add the variable to a text object. Add as many text boxes as there were fields in the Database Query action and add the same variable name you assigned to the field in the text box.

6: You can format each text box just like any other, you may want to add a border or change the font size. You can add transitions and effects or show and hide the text field as you would with any text object.

7: That's it! How it works is that when the page is displayed, the On Show activates the Database Query Action that then connects to the database and creates a list of records that match the criteria you set.

Next it gets the first record from the list and adds the contents from each of its fields to the variables that you set up to store the data. Finally, it displays the contents of the variable in the text Objects in which you inserted the variable name.

Making databases interactive

What if you want to go to the next record or go back to the very first record? There are a variety of Database Actions that let you interact with the database (see *Actions: Database* for more information).

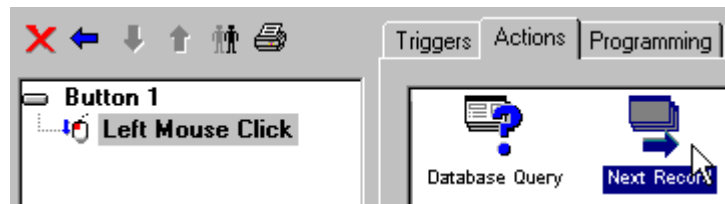
The following steps will show you how to create a 'Next Record' button, which will get the next record from the database whenever you click on the button. You can use the same steps to create 'Previous Record', 'First Record' or 'Last Record' buttons:

Create a Next Record button:

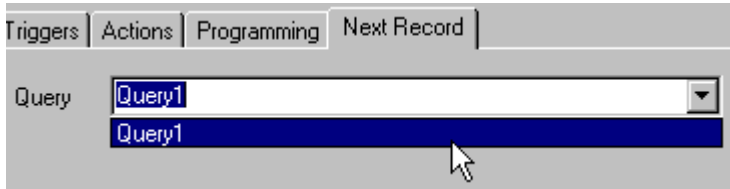
1: Create a new button on the same page as the text boxes containing the record information appear, and then open the buttons Actions dialog. The Left Mouse Click trigger is automatically selected for buttons so you won't have to add it.

2: Once you have selected the trigger that will start the process, select the Database tab from the list of mini tabs at the bottom of the Actions tab

3: Double-click on the **Next Record** icon to add it to the Left Mouse Click trigger. A Next Record tab will appear after the Programming tab in the Actions dialog, this is used to select the query, which is the list of records of which you want to show the next record.



4: In the Next Record tab, select the same query name as you gave to the query when you originally created the query in the Database Query action.



A list of all the queries currently created in your publication will appear in a drop down list from which you can select your query.

5: And you have finished. How it works is that when the user clicks on the button, the Next Record action is activated. The system will look up the query you entered in the Next Record tab and find the next record in the list. It will then update the variables displayed on the page with the data from the fields of the next record. Each time you press the button, the next record in the query is displayed. If the current record displayed on the page is the last record in the query then nothing will happen.

Connect to a Database

For each database you want to use in an Opus publication, you must create a database connection. To make the connection, Opus uses the Microsoft Open Database Connectivity drivers (more commonly known as ODBC drivers) to interface between the database and Opus.

To connect to a database, the ODBC drivers must be installed on the development machine. In Windows 95, 98, NT and 2000, the ODBC drivers should already be installed on your machine. To find out which drivers are currently installed on your Windows system:

For Windows 95, 98 or NT:

- 1:** From the desktop select Start > Settings > Control Panel.
- 2:** Double-click the 32bit ODBC icon to open the ODBC administration dialog. Alternatively, the icon may be named ODBC Data Sources or just ODBC.
- 3:** Click on the Drivers Tab to view the current drivers installed on your system.

For Windows 2000:

- 1:** From the Windows desktop select Start > Settings > Control Panel.
- 2:** Double-click the Administrative Tools folder and then double-click the Data Sources (ODBC) icon to open the ODBC administration dialog.
- 3:** Click on the Drivers Tab to view the current drivers installed on your system.

If the ODBC icon is not present on your system, it is most likely that ODBC has not been installed on your machine. You can load ODBC drivers from your original Microsoft Windows CD-ROM.

Data Source Names (DSN's)

There are a variety of ODBC drivers, one for each type of database system. For example, there are individual ODBC drivers for, Access,

FoxPro, Paradox, dbase, Oracle, and so on. However, the function of each ODBC driver is the same. ODBC drivers translate the data from your database in a meaningful way so that another program, such as Opus, understands what information it is receiving. Therefore, each database you intend to use in Opus is given a Data Source Name (DSN).

The DSN is simply a combination of the name of the database you want to open in Opus and the appropriate ODBC driver that will be used to interpret the data for Opus. For example, if you have two databases both created in MS Access called 'login' and 'catalogue' you will need to create two DSN's, one for each database. Each DSN will state that the Access ODBC driver as the driver used to interpret the data.

Creating a new DSN

To create a new DSN in Opus:

- 1:** Select the object that will open the database. This is most commonly a button, an image or a page.
- 2:** Double-right click on the object to open its Actions Editor.
- 3:** Select the Action Trigger that will open the database e.g. 'Left Mouse Click' trigger for a button or an 'On Show' trigger for a page.
- 4:** Select the Actions Tab and scroll through the Action tabs at the bottom of the window to the Database Tab.
- 5:** Click and drag the **Database Query** icon from the right-hand column to the Action Trigger in the left-hand column. A new Database Query Tab will appear beside the Programming Tab in the right-hand column.
- 6:** Click the **Select** button in the Database Query Tab to open the 'Select Data Source' dialog.
- 7:** Click the **New** button in the 'File Data Source' Tab to start the **Create New Data Source Wizard**.
- 8:** Select the appropriate driver from the list and click on the 'Next' button to continue e.g. to connect to an Access database select 'Drive do Microsoft Access (*.mdb).
- 9:** Type an appropriate name for this database connection and press the 'Next' button to continue e.g. if the database is a catalogue of your

products, then type 'catalogue' as the name. This name is the DSN (Data Source Name), which describes to your system the type of database you are trying to connect to.

10: Click on the **Finish** button to close the Wizard. A new setup dialog will automatically appear in which you select the database file you are trying to connect to Opus.

11: Click the **Select...** button in the setup dialog to open the **Select Database** dialog.

12: Navigate to the folder that contains your database file and select it, then click on the 'OK' button to close the 'Select Database' dialog.

13: Click the 'OK' button to close the setup dialog. The original 'Select Data Source' window will re-appear with your new DSN connection in the list e.g. if you called your new DSN catalogue then 'catalogue.dsn' would now appear in the list of connections.

14: Make sure your new connection is in the field 'DSN Name' and click the 'OK' button to return to the 'Database Query Tab' in Opus.

Re-using DSNs

Once you have created a new DSN connection for a database on your system it is permanent. You can re-use the same DSN in other Opus publications or in the same publication on other pages. When you want to open the database in a new publication, choose the DSN from the list provided when you press the 'Select...' button in the 'Database Query Tab'.

To create a query in Opus, you must use the **Database Query** Action from the Database tab in the Actions dialog. A Query is simply a method of extracting particular records from a database that you want to display within an Opus publication (see Database Query in *Actions: Database* section).

Creating a Database Query

A standard method of creating a query is to write an SQL statement. SQL stands for Structured Query Language, it is a standardized language that selects and rejects records from a database depending on whether they match the criteria you outline in the statement. SQL statements can be complicated to construct so Opus has provided an SQL Wizard that will generate a query for you by simply filling in boxes.

The SQL Wizard

In Opus the only place you need to create a query is in the **Database Query** Action and this is where you will find the SQL Wizard (see Database Query in *Actions: Database* section).

To start the wizard, click on the SQL Wizard button in the Database Query tab.



The SQL Wizard is made up of five screens, as follows:

The first screen contains one entry:



1: Tick the check box if you intend to use either the **Update Current Record** Action or **Delete Current Record** Action. Clear the check box if you do not wish to update or delete new records within Opus. Click on the 'Next' button to continue.

Note: This option does not affect the **Insert New Record** action.

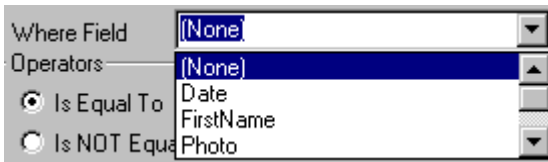
2: On the second screen, select the required table from the list of tables available for that database. The table's field names are automatically displayed below the selected table.



3: Place a tick in the check box for each field you want to display in the Opus publication. If your publication does not require information held in a field, clear its check box.

4: Click on the **Next** button to continue.

The third screen is used to set the selection criteria of your query i.e. which records from the table will be part of the query.



5: Select the field name from the drop down list of the **Where Field**. The list will contain all the fields from a table whether you ticked them on the previous screen or not, this is because you may want to use a field as a comparison but not show it in your publication. This field is the comparison field, in other words, the data contained in each record of this field is compared with the 'Value' or 'Variable' set in the 'Compare To' section.

TIP: To select ALL the records in the current table simply leave the 'Where field' set to [None] and click on the 'Next' button to continue. When you select All the records, the fourth screen in the Wizard will not be displayed.

Operators

Is Equal To Is Greater Than

Is NOT Equal To Is Less Than

Is LIKE Is Greater or Equal To

Is NOT LIKE Is Less or Equal To

Is NULL Is NOT NULL

6: Next select the required Operator from the list. An Operator is a method of comparing the value of the ‘Where Field’ with the value of ‘Compare To’ to test if it should be included within the current query. Most of the operators are self-explanatory, for example, ‘Is Equal to’ would be used when you wanted the contents of the records to exactly match the ‘Value’ or ‘Variable’ entry, such as finding all the records with the department name ‘Support’ and the ‘Is Greater Than’ operator can be used to find records above a certain value, such as all ages that are greater than 20.

The LIKE field needs a little more explanation and is described in the *Using the LIKE Operator* section below.

Compare To

Value

Variable

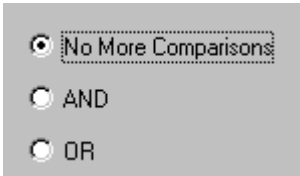
7: To check if a record should be added to the query set the ‘Compare To’ section to either a ‘Value’ or ‘Variable’.

- A ‘Value’ is a literal value i.e. any letter, character, word or phrase you enter in the ‘Value’ field e.g. 23, SMITH, John Doe.
- A ‘Variable’ is any global or page variable created by the system or by the user e.g. SCORE_TOTAL or ‘User_Name’.

8: Click on the **Next** button to continue.

The fourth screen allows you to add another selection criteria to your query, for example, you may want to find all prices above £30 AND all prices below £100; alternatively, you may want to find all employees in the Sales department OR the marketing department.

9: There are three options available on this page of the Wizard:



No More Comparisons – means that you do not want to add another query to your SQL statement. Selecting this option and pressing the Next button will move immediately to the fifth page of the Wizard.

AND – means that you want to add another query to your SQL statement. Selecting this option and clicking the Next button will return to the previous page of the Wizard where you can then add another selection criteria. When you use the AND comparison, it means that in order for a record to be added to the Select List (i.e. the list of returned records) the record must match both the first selection criteria AND the second.

OR – also means that you want to add another query to your SQL statement. Again, by selecting this option and clicking the Next button, you will return to the previous page of the Wizard where you can then add another selection criteria. How this differs from the AND comparison is that the record will be added to the Select List if it matches either the first selection criteria OR the second criteria.

The fifth and final screen allows you to set the sort order for your query.

10: Select the name of the field by which you want to sort the records e.g. for a table containing product details you will probably sort the table by the product name field in Ascending order.

The image shows a 'Sort By' dialog box with three sections. Each section has a 'Field' dropdown menu and two radio buttons: 'Ascending' (selected) and 'Descending'. The first section has '(None)' in the dropdown. The second and third sections also have '(None)' in the dropdown.

Records can be sorted by more than one field, for example, records that are initially sorted by a surname field could then be sorted by a forename field. Records can be sorted by a maximum of three fields.

11: Click on the **Finish** button to return to the Database Query tab.

Using the LIKE Operator

The LIKE Operator is one of the ways in which you can select records from a database using the SQL Wizard. This operator allows you to enter a word or a phrase that must be found in the field of a record to be included in the query. However, the LIKE operator also allows you to match any single character or a range of characters from a list of characters that you supply. For example, you could find all records with the surname 'Smith', 'Smyth' or 'Smythe' by using the LIKE operator. There are two special characters that you can use to indicate a single character or a range of characters in your comparison.

Use the underscore character (i.e. the `_` symbol) to indicate a single character.

Use the percent character (i.e. the `%` symbol) to indicate a range of characters

In the examples below, the Pattern is the text you would enter in the 'Compare To' value or variable that is being used in the comparison page of the SQL Wizard:

Pattern	Meaning
d%	The letter d followed by 0 or more characters e.g. drip, drown, dog, date.
_n	Any single letter followed by n e.g. an, in, on.
Sm_th%	Will find Smith, Smyth and Smythe as well as Smiths, Smithson, and so on.

You can also find a range of characters by surrounding the characters with square brackets (i.e. the [] symbols). Just like the previous examples, the examples below can be used in the 'Compare to' value or variable that is being used in the comparison page of the SQL Wizard. You can also combine the square brackets with the underscore and percent characters:

Pattern	Meaning
[A - F]	Will find all characters from the range A to F.
[ASDF]	Will match any of the characters A, S, D or F.

Matching Dates and Times

One other thing you should be aware of when setting up your query in the SQL Wizard is finding records by date or time. You should use the following formats for timestamps, dates and times:

Timestamps	2001-12-31 12:00:00
Date	2001-12-31
Time	12:00:00

These are standard formats for extracting records from date and time fields.

Actions: Database

This section of the Actions dialog provides you with controls for showing the records in a database within an Opus publication. You can quickly connect to an external Database, set a Query to extract data from the database and easily navigate to other records in the database using the actions provided.

Opus can connect to a wide range of databases without the need to learn Structured Query Language (SQL), a common method of interacting with a database. This level of complexity is handled by Opus so you can get on with the task of providing stimulating and engaging publications.

Database Query



This Action is used to:

- Setup the connection with a Database
- Select the records from the database by creating a Query
- Set the field names in the database to variables in Opus

At least one Database Query must be set up for each database you want to use in an Opus publication.

When you select this option a fourth ‘Database Query’ tab will appear after the Programming tab in the Actions dialog.

To setup a Database Query:

1. Enter the DSN (Data Source Name) in the DSN field – the DSN is the method of connecting to a database (see *Connecting to a Database* section for more information). Click on the Select button to create a new DSN or select a previously entered DSN from the drop-down list box to the right of the field.



The image shows a graphical user interface element for setting a Data Source Name (DSN). It consists of a rectangular box with a light gray background. On the left side of the box, the text 'DSN' is displayed. To the right of 'DSN' is a white text input field. To the right of the input field is a small, dark gray square containing a white downward-pointing triangle, indicating a drop-down menu. To the right of the drop-down menu is a button with a light gray background and the text 'Select...' in a dark font.

2. Enter a new Query by clicking on the Query Wizard or select a previously entered Query from the drop-down list to the right of the field. A Query sets which records from the database you want to be able to read into Opus (see *Creating a Database Query* section for more information).



This field will automatically create a new name for the Query ID, the first time you create a query, the ID will be 'Query 1', the second time 'Query 2', and so on. However, we suggest you make the name more descriptive, such as 'Product range' or 'All employees'. To enter your own Query ID, simply type in the name before you press the SQL Wizard. When you have completed your selections in the SQL Wizard, the SQL statement will appear underneath the Query ID field, this statement can only be changed by clicking on the SQL Wizard button again.

3. Once you have created a new Query, the table at the bottom of the tab will contain a list of fields for the selected table in the Query. To view the data contained in a field you must assign it to an Opus variable, in other words, you won't see any data on a page in Opus unless the contents of the field are saved into a variable.

Field	Variable
Prod_id	(None)
Prod_name	(None)
Short_desc	(None)
Long_desc	(None)
Image	(None)

To assign a field to a variable, simply select the field name in the Field column and the Change button at the bottom of the pane will highlight. Click on the Change button and the familiar variable dialog box will appear in which you can select a variable from the list or create a new variable name. Give each field name in the Field column a different variable name (remember to make the variable name's descriptive).

4. Click on the Apply button to save your settings and click the OK button to shut the Actions dialog.

Showing database records on a page

Setting up the Database Query is the hard part, once you have set up the Action it is simple to show the records on the screen.

All you need to do is draw a text box on the page for each field in the database and insert the variable name you have assigned to the field. Whenever the Database Query action is triggered, for example, when the page is displayed the Database Query will do the following:

- First it will connect to the database
- Next it will run the SQL Query and find the first record that matches the query.
- Finally, it will get the contents of each field in the record and insert it into the variable name that has been assigned to it

Next Record



This Action will select the next record from a query and replace the current value of each field variable with the field contents of the next record. For example, if the table contained two fields (Forename and Surname) and the field variables assigned to them were 'Fname' and 'Sname' respectively, the 'Next Record' Action will clear the current value of the field variable 'Fname' and 'Sname' and replace them with the contents of the next records Forename and Surname field. Please note, if the current record is the last record in the query, the field variables will remain unchanged.

Select the query name from the 'Query' field drop down list and click the 'Apply' button to save your settings. A 'Next Record' Action is normally placed on a button object.

Previous Record



This Action works in the same way as the 'Next Record' Action but replaces the current value of each field variable with the field contents of

the previous record in the Query. Please note, if the current record is the first record in the query, the field variables will remain unchanged.

Select the query name from the 'Query' field drop down list and click the 'Apply' button to save your settings. A 'Previous Record' Action is normally placed on a button object.

First Record



This Action will jump back to the first record in the query from its current position. The current value of each field variable will be replaced with the field contents of the first record in the query.

Select the query name from the **Query** field drop down list and click the Apply button to save your settings. A **First Record** action is normally placed on a button object or triggered by an **On Show** trigger on a page.

Last Record



This action will jump to the last record in the Query from its current position. The current value of each field variable will be replaced with the field contents of the first record in the Query.

Select the query name from the **Query** field drop down list and click the **Apply** button to save your settings. A **Last Record** Action is normally placed on a button object.

Delete Record



This action will delete the current record in the query. Please note, this Action will not automatically ask for confirmation prior to deleting the record.

Select the query name from the 'Query' field drop down list and click the 'Apply' button to save your settings. A 'Delete Record' Action is normally placed on a button object. To ensure a record is not

accidentally deleted from a database, we suggest you create an 'If' or 'Loop' Action that requires a user response of 'Yes' or 'No' before using the 'Delete Record' Action.

Note: A record can only be deleted if the 'Select data for Updating/Deleting' option was checked when the Query was initially created in the SQL Wizard. If this option was not checked, this action will do nothing.

Update Current Record



This Action is used to update the current record in the Query with new information. As with the 'Delete Record' Action, this Action will not automatically ask for confirmation prior to updating the current record. An 'Update Current Record' Action is normally placed on a button object.

When you select this action, an 'Update Current Record' tab will appear after the Programming tab in the Actions dialog. To update the current record:

1. Select the query name from the 'Query' field drop down list. A list of field names contained in the table will appear in the 'Field' column below the Query field.



Query

2. Highlight a field from the 'Field' column...

Field	Set To
UserID	<UserID>
FirstName	<FirstName>
Surname	<Surname>
Date	<Date>
Photo	<Photo>

... and select the required radio button from the list below.

There are four choices and only one can be selected for each field:

String – Any letter, number, word or phrase can be entered in the String field.

Variable – Select a name of a variable from the Variable drop down list. The value contained in the variable will be added to the field and not the name of the variable. For example, you could have a user update a record by creating an insert text box on the page that saved the user's entry in a variable; that variable could then be entered in this field.

Null – The field will contain nothing, in other words, if the field currently had an entry it would be replaced by a blank entry.

Don't Update – This option will not update the field.

3. Repeat step 2 for each field you want to update in the current record. Please note, you are not required to update all fields; any field that you do not update will retain its current data.

4. Click on the 'Apply' button to save your settings.

Note: A record can only be updated if the 'Select data for Updating/Deleting' option was checked when the Query was initially created in the SQL Wizard. If this option was not checked, this action will do nothing.

Insert New Record



This Action is used to insert a new record in a database. The data for each field of a new record can be set via a 'string' of text, an Opus variable or the database default entry for the field as well as to 'Null' (i.e. no entry).

When you select this action, an 'Insert New Record' tab will appear after the Programming tab in the Actions dialog. To insert a new record:

1. Enter the DSN of the database to which the new record should be added.



Alternatively, you can create a new DSN by clicking on the Select button.

2. Select the required table from the drop-down list box that appears to the left of the Table field. A list of field names contained in the table will appear automatically in the 'Field' column below the Table field.



3. Highlight a field from the 'Field' column...

Field	Set To
UserID	
FirstName	
Surname	
Date	
Photo	

...and select the required radio button from the list below:



There are four choices and only one can be selected for each field:

String – Any letter, number, word or phrase can be entered in the String field.

Variable – Select a name of a variable from the Variable drop down list. The value contained in the variable will be added to the field and not the name of the variable. If you want users to create new entries in a database, you could create a new 'Records' page in your publication. For each field in the database, draw an Insert Text box on the Records page and set the Input Text tab in its Properties dialog to save the user's entry to a variable, then simply enter the name of that variable in this field.

Null – The selected field will contain nothing.

Don't set – This option allows the database in which this new record will be inserted to determine the entry for this field.

4. Repeat step 3 for each field listed in the **Field** column.

5. Click the **Apply** button to save your settings.

Note: A new record can only be inserted if the 'Select data for Updating/Deleting' option was checked when the Query was initially created in the SQL Wizard. If this option was not checked, this action will do nothing.

End Query



This Action will close the connection to the database. You should use this Action when you no longer want to use the database within your publication because it will free up memory on your system.

Select the query name from the 'Query' field drop down list and click the 'Apply' button to save your settings. All queries are associated with a single database, therefore this Action will only close the connection for the database associated with the named Query.

Number of Records



This Action will count the number of records in the Select List of a Query; the number is then saved in a variable. Please note, this Action will not return the total number of records in a Table, only the total number of records in a Query. In other words, if the Select List of a Query is a sub-set of a Table (i.e. the Query does not select all the records in a Table) the number returned is the total number of records in the sub-set.

When you select this action, a 'Number of Records' tab will appear after the Programming tab in the Actions dialog. To set this Action:

1. Select the query name from the drop-down list box that appears to the



right of the Query field.

2, Select the variable name from the drop-down list box that appears to the right of the Variable field...

A screenshot of a software interface. On the left, the word "Variable" is followed by a text input field. To the right of the input field is a small downward-pointing arrow icon, indicating a drop-down menu. Further to the right is a button labeled "New...".

... or click on the 'New...' button to create a new variable.

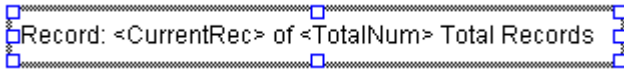
3. Click on the 'Apply' button to save your settings.

Tip: This action is normally used with the 'Current Record Number' action to display information to the user in a format similar to:

Record: 1 of 8 Total Records

This can be easily achieved by drawing one text box on the page and inserting the variable names in which you have saved the 'Number of Records' and 'Current Record Number' actions.

For example, if you had named the variables as: <CurrentRec> and <TotalNum>, the entry in the text box would look like this:



Record: <CurrentRec> of <TotalNum> Total Records

Current Record Number



This Action will return the position number of the current record in the Select List of a Query; the number is saved in a variable. For example, if a Query contained ten records, the position number of the first record in the query is 1, the second 2, and so on. A 'Current Record Number' Action is normally placed on a page and activated by an 'On Show' Trigger, the variable is then displayed in a Text object.

When you select this action, a 'Current Record Number' tab will appear after the Programming tab in the Actions dialog. To set this Action:

1. Select the query name from the drop-down list box that appears to the



Query

right of the Query field.

2. Select the variable name from the drop-down list box that appears to the right of the Variable field...



Variable

New...

... or click on the 'New...' button to create a new variable.

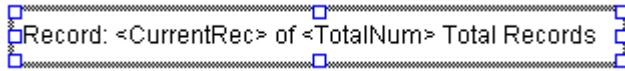
3. Click on the 'Apply' button to save your settings.

Tip: This Action and the 'Number of Records' Action are often used together as information fields for users. For example, one Text object could contain the variable names used in the 'Number of Records' and 'Current Record Number' Actions to display:

Record: 5 of 10 Total Records

This can be easily achieved by drawing one text box on the page and inserting the variable names in which you have saved the 'Number of Records' and 'Current Record Number' actions. For example, if you had

named the variables as: <CurrentRec> and <TotalNum>, the entry in the text box would look like this:



Record: <CurrentRec> of <TotalNum> Total Records

The Resource Manager

The Resource Manager...43

The Resource Manager

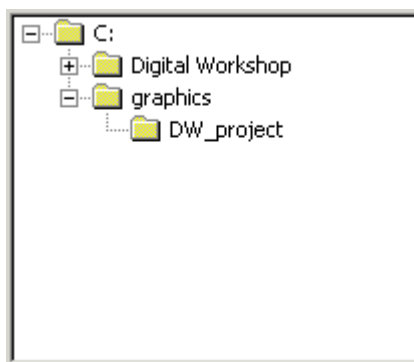
The Resource Manager is an incredibly powerful tool that allows you to review, update, move and delete resources in a publication. Managing your resources when developing your publication is very important and this tool provides a one-stop shop for all your resourcing needs. You can even set or reset publishing options from the Resource Manager, prior to publishing.

To open the Resource manager, click on the **Tools** menu at the top of the screen and select **Resource Manager**. The Resource Manager window will open.

Resource Manager Workspace

The Resource Manager window is made up of three elements:

The **Organiser** on the left provides an Explorer-style view of the directories in which your publication resources are currently held. In this section you can open and close folders and sub-folders by clicking on the plus and minus buttons.



The main pane to the right of the Organiser provides a list of resources in the currently selected folder in the Organiser.

Filename ▲	Size	Mode	Object	Page
fire.jpg	195KB	Default	Button 1	Page 2
		Default	DW Project	
frogman.jpg	124KB	Default	DW Project	
gasmask.jpg	147KB	Default	DW Project	
heli_moon.jpg	125KB	Default	DW Project	
		Default	Image 2	Splash
image2.jpg	161KB	Default	DW Project	
image4.jpg	141KB	Default	DW Project	
intro.jpg	98.2KB	Default	DW Project	
		Default	Image 1	Page 1

For each resource displayed, you can find out its: Filename; Size in Kilobytes; Mode of operation (normally default); the name of the Object that uses the resource, that is, the name of the image or button in which the resource is used; and finally, the page on which the resource is being used.

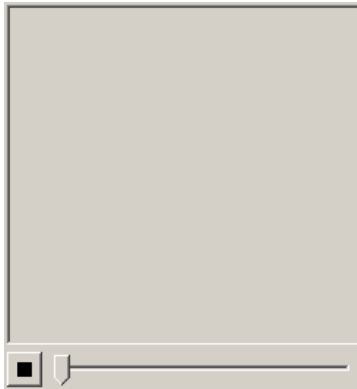
When a resource is used more than once, then all its uses are displayed in the list. In the illustration above, the fire.jpg is used in 'Button 1' on 'Page 2' and has been added as an additional resource to the 'DW Project'.

The order in which the resources are displayed by default is in ascending alphabetical order of filename, as indicated by the up arrow beside the **Filename** column heading.



You can arrange the resources by any of the column headings by simply clicking once on the column heading; the arrow will be placed alongside the column heading and the resources will be re-ordered. If you click on the arrow itself, then the order will be reversed from its current position.

The third element in the Resource Manager window is initially hidden and can be opened by pressing the **Preview** button in the bottom right-hand corner of the window. This will open the Preview pane showing a Resource player that can display an image or play a video, animation or sound.



To close the Preview pane simply click on the **Preview** button again and the pane will be hidden.

The Toolbars

The Resource Manager contains one toolbar. Each toolbar button has a tooltip, so if you linger over the button it will remind you what it does.



The toolbar is split into different functions as described below:



These tools control the options displayed in the Organiser and the resources displayed in the main pane to the right of the Organiser.

Choose by folder - is the first option and will display all folders on your system that contain resources used in your publication.

Choose by page - will list all the pages in your publication in the Organiser. Selecting a page from the list will display all the resources used in the page in the main pane.

Choose by chapter - will list all the chapters in your publication in the Organiser. Selecting a chapter from the list will display all the resources used in the chapter in the main pane.

Choose by component - is the final choice. This option is greyed-out until you select a Component Gallery from the main Opus Organiser. When you have selected a component gallery, this option will be enabled.

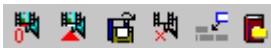


These tools control the resources displayed in the main pane to the right of the Organiser.

Show Resources In Subfolders - is the first option and will display all the resources in the currently selected folder in the Organiser as well as resources in its subfolders. If the button is indented, then the option has been selected.

Show All Uses - a resource can be used more than once in the same publication, the main pane will show each instance a resource has been used. If you do not want to see every instance of a resource, make sure the button is not indented.

Filter Resources - this button will open the Filter Resources dialog box in which you can select the type of resources to display in the main pane, such as all files, missing files, present files or specific File Types (see *Filter Resources* section).



These are the management tools that offer a variety of dialog windows in which you can move, update, consolidate resources and much more.

Set\Clear Resources this will open the Set\Clear Resources dialog in which you can clear specific resources from the publication or set resources to a file (see *Set\Clear Resources* section).

Consolidate Resources - will open the Consolidate Resources dialog in which you can copy all or some of the resources in the publication into a resource subdirectory (see *Consolidate Resources* section).

Change Resource Path - will open the Change Data Paths dialog in which you can move resources from one folder to another (see *Change Data Path* section).

Purge Resources - will open the Purge Resources dialog in which you can remove resources from a publication that are not currently used or create a purge list to select individual resources that you may want to remove at a later time (see *Purge Resources* section).

Rebuild Resources - will instantly update the Resource Manager. For larger publications this may take some time.

Publish Location - will open the Publish Options dialog in which you can change a variety of publishing settings, such as the location of the publication, image and file compression (see *Publish Location* section).

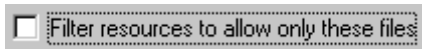


The final two buttons on the **Resource Manager** toolbar are the **Undo** and **Redo** buttons. Any operation you perform within the **Resource Manager** can be undone. The **Undo** and **Redo** buttons keep a history of the actions you have performed in the **Resource Manager**, which means you can undo not only your last action, but every action you performed, if required.

Filter Resources

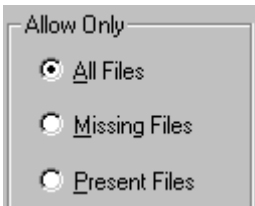
The Filter Resources dialog is used to control which resources should be displayed in the main pane of the **Resource Manager** window; the resources are not removed from your publication, they are just temporarily hidden. You should only use this option if you want to hide (i.e. filter) particular resources from the list displayed in the main pane, such as to display only the missing resources in your publication.

1: Place a tick in this box if you want to hide particular resources from the **Resource Manager** window's main pane.



Note: if this option is not ticked, then all the resources in the publication will be displayed, this is the default setting when you first enter the **Resource Manager**.

2: Select the type of resources you want to display in the main pane. There are three options:

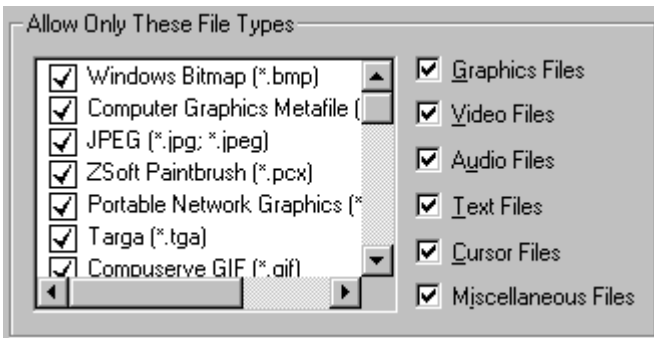


All Files - is the first option, it will display all resource files in your publication whether or not they are missing or present.

Missing Files - will only display the resource files that are missing. If some of your objects or pages go blank it may be because the resource file is missing. This option will find all the missing files and display them in the main pane of the **Resource Manager**. You can then work out the reason why they are missing – it may be that the resources were on a CD that has been removed or in a network drive that is currently unavailable.

Present files – does the opposite of the Missing Files option, in other words, it will only display resource files that the system can locate.

3: The final section of the **Filter Resources** dialog allows you to further refine the resources you want to display in the main pane by selecting certain File Types:



By default all File Types are selected, which is indicated by the tick beside the File Type name. In the left-hand box is a list of all File Types normally found on a Windows PC. In the right-hand column the various File Types are grouped into categories, for example, Graphical Files would include (bmp, jpg, gif, png and a host of other File Types associated with images).

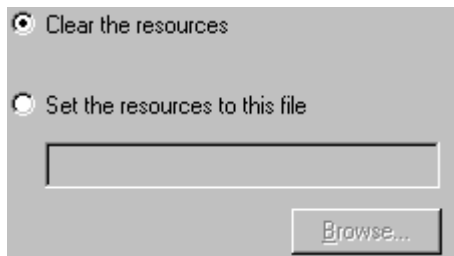
If you want to only display a certain category, such as Graphical Files and Video Files, you can quickly hide all other File Types by removing the tick from all the other categories except Graphical Files and Video Files; in the left-hand box, only the File Types associated with these categories will remain ticked. You can further refine the filter by then removing the ticks from specific File Types in the left-hand box.

4: Once you have made your selections, click on the OK button to return to the **Resource Manager** window.

Set\Clear Resources

The **Set\Clear Resource** dialog is used to either remove resources from your current publication or change one resource for another within your publication.

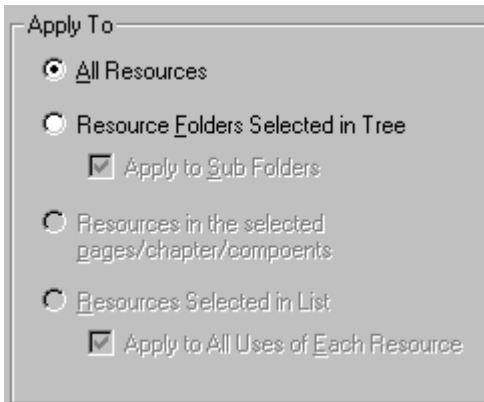
1: First decide if you want to clear resources or set resources to a file:



Clear the resources – click on this option and you will remove resource files from the publication. If the resource is being used by an object, such as an image in an image object, the object remains but the image is removed.

Set the resources to this file – click on this option and you will change existing resources for a new resource. This is very useful if you have the same image used on 50 different pages in your publication because you can set the image to a new resource file name and the new image replaces the old image on all 50 pages. To select the new file, click on the **Browse** button and locate your new file.

2: Once you have decided if you want to clear or set resources you have to decide which resource files this will affect. You have the following options:



All Resources – this option will clear or set all the resources currently in your publication.

Resource Folders Selected in Tree – this option will clear or set all resources in the folder currently selected in the **Organiser** of the **Resource Manager** window. You can also clear or set the resources in subfolders of the current folder if you tick the **Apply to Sub Folders** option.

Resources in the selected pages/chapters/components – this option will clear or set all resources in pages, chapters or components that are currently selected in the **Organiser** of the **Resource Manager** window. This option is only available when you have displayed resources using the **Choose By Page**, **Choose By Chapter** or **Choose By Component** button.

Resources Selected in List – this option will clear or set only the resources that have been selected in the main pane to the right of the **Organiser** in the **Resource Manager** window.

3: Finally, you can further refine the list of resources that will be cleared or set by adding a filter:



Tick the **Use Filter** option if you want to clear or set resources for a specific File Type, for example, to clear all Video File formats but not Graphical File formats. If you have not previously set up a filter, click on the **Set Filter** button to open the Filter Resources dialog.

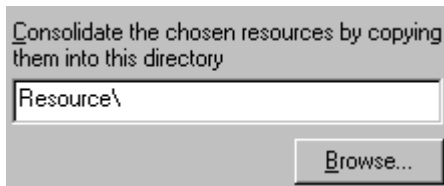
4: Once you have made your selections, click on the OK button to start the clear or set procedure.

Consolidate Resources

The **Consolidate Resources** dialog is used to copy *selected resource files* used in the publication into a Resource subdirectory of wherever you saved your publication (or the Windows Temp directory if you haven't yet saved the publication).

Note: This dialog allows you to copy all resource files or *selected resource files* that you want to add to the Resource file (see point 4 below for more information).

1: Select the subdirectory in which the resources should be copied:



You can specify a different subdirectory by clicking on the **Browse** button and selecting a new location.

2: You can merge duplicate resource files, which means that if you have used the same resource in several places within your publication, the Resource directory will keep just one copy.

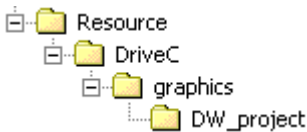


Tick the box if you want to merge files or remove the tick if you don't.

3: The Preserve Directories option allows you to create subdirectories within your Resource folder, which may be useful if you want to group all images in a graphics folder or videos and animations in a movies folder.



Tick the Preserve Directories option to keep directories. For example, if the resources are currently in the directory 'c:\graphics\DW_project' and you want to preserve this directory structure in the Resource folder, the Resource folder will look something like this if viewed in Windows Explorer:

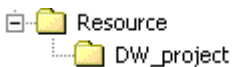


Tick the Do not preserve the drive letter option if you do not want to include the disk drive letter as a folder in the resources. In the illustration above, the folder DriveC would be removed.

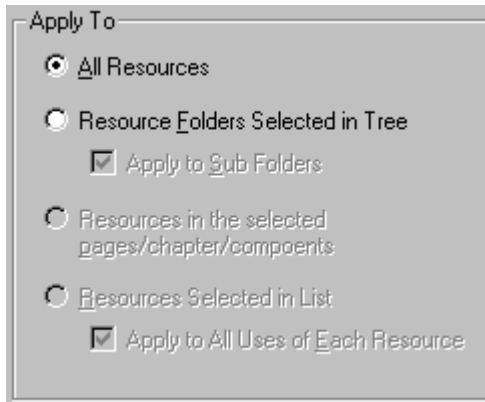
Finally, you can state the number of directories that should be removed:



Tick this option and enter the number of subfolders that should be removed when resource files are added to the Resource directory. To use the example above, if you enter 2, the Resource folder would look like this:



4: Normally, all resources are added to the Resource folder when you Consolidate Resources. However, you may want to add certain selected resource files. You have the following four options available to you:



All Resources – this option will consolidate all the resources currently in your publication.

Resource Folders Selected in Tree – this option will consolidate resources in the folder currently selected in the **Organiser** of the **Resource Manager** window. You can also consolidate resources in subfolders of the current folder if you tick the **Apply to Sub Folders** option.

Resources in the selected pages/chapters/components – this option will consolidate all resources in pages, chapters or components that are currently selected in the **Organiser** of the **Resource Manager** window. This option is only available when you have displayed resources using the **Choose By Page**, **Choose By Chapter** or **Choose By Component** button.

Resources Selected in List – this option will consolidate only the resources that have been selected in the main pane to the right of the **Organiser** in the **Resource Manager** window.

5: Finally, you can further refine the list of resources that will be consolidated by adding a filter:



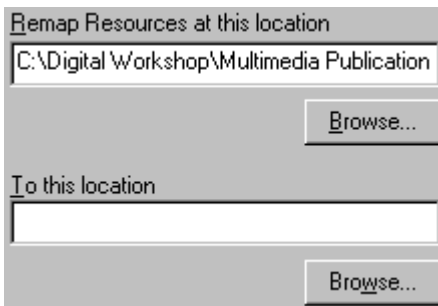
Tick the **Use Filter** option if you want to consolidate resources for a specific File Type, for example, to consolidate all Video File formats but not Graphical File formats. If you have not previously set up a filter, click on the **Set Filter** button to open the Filter Resources dialog.

6: Once you have made your selections, click on the OK button to start the consolidation process.

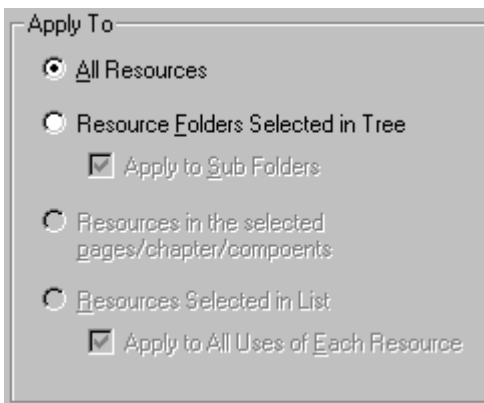
Change Data Paths

Some users will want to access their publications on different network drives, or colleagues may work on them on different workstations. The **Change Data Path** dialog allows you to quickly change the data drive for resources.

1: Enter the new location in the **To this location** field to re-direct Opus to the new directory where the resources may be found.



2: Normally, all resources are remapped to the new location when you change data paths. However, you may want to choose specific resource files that you want to remap. You have the following four options available to you:



All Resources – this option will remap all the resources currently in your publication.

Resource Folders Selected in Tree – this option will remap resources in the folder currently selected in the **Organiser** of the **Resource Manager** window. You can also remap resources in subfolders of the current folder if you tick the **Apply to Sub Folders** option.

Resources in the selected pages/chapters/components – this option will remap all resources in pages, chapters or components that are currently selected in the **Organiser** of the **Resource Manager** window. This option is only available when you have displayed resources using the **Choose By Page**, **Choose By Chapter** or **Choose By Component** button.

Resources Selected in List – this option will remap only the resources that have been selected in the main pane to the right of the **Organiser** in the **Resource Manager** window.

3: Finally, you can further refine the list of resources that will be remapped by adding a filter:



Tick the **Use Filter** option if you want to remap resources for a specific File Type, for example, to remove all Video File formats but not Graphical File formats. If you have not previously set up a filter, click on the **Set Filter** button to open the Filter Resources dialog.

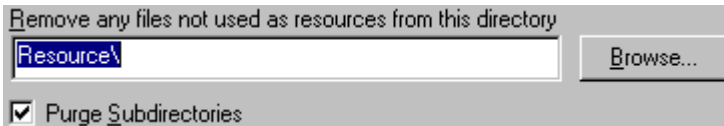
4: Click on the **OK** button to change the data paths.

Purge Resources

When you have completed your publication and are confident that you no longer require any unused resources, you can use the **Purge Resources** dialog to remove unused resources from your publication. Alternatively, this dialog contains an option to **Make Purge List** that will contain a list of all resource files in the publication from which you can select specific resources to remove.

1: Decide whether you want to purge all resources or **Make Purge List**.

To purge all unused resources:



When you first open this dialog, this option is automatically selected and the current name of the Resource directory for this publication is entered in the field. There is only one other option available when purging all resources and that is the **Purge Subdirectories** option. Tick this option if you want to remove all unused resources in subdirectories of the Resource directory. Click the **Do Purge** button at the bottom of the dialog to begin the purge.

To purge a selected range of resources:



Click on the Make Purge List button – this will list all unused resources in the **Files to be purged** pane underneath the button. The rest of the options on this screen are used to select which resources you want to purge.

2: Select the resources from the **Files to be purged** pane that you want to remove from the resources directory. You can select a single resource or any number of resources contained within the list.

3: Once you have selected the resource files to be removed, select one of the following options:



Delete the files – this option will delete the resource files from the system completely, they cannot be recovered at a later date.

Move the files to the Recycle Bin – will move the resources to the Windows Recycle Bin, which you can then delete as and when you want to. You can also take files out of the Recycle Bin if you decide you need them after all.

Move the files to this directory – use the Browse button to select a directory where you want to move the resource files. This option is useful if you are not sure whether you may need some of the resources at a later date.

4: The final option you have is to delete any subdirectories when it contains no files.

Delete Empty Subdirectories

The box is automatically ticked to remove empty subdirectories. Remove the tick if you want to keep subdirectories, even if they contain no files.

5: Click on the **Do Purge** button to start the purge.

Additional Features

Arrays...	61
ListBoxes and Tables...	65
Layout...	71
Publication Properties...	75
Additional Launch Actions...	89
Clock Action...	93
File Management Actions...	97
Conditional Actions...	103
Information Tab...	107
Permissions...	109
Using Parameters...	85

Arrays

An **array** is like a series of variables. It can hold different individual pieces of information and you can retrieve the pieces of information individually. These segments of information are called the **elements** of the array. You access each element by its number in the array. This means you can store multi-level information and access pieces of information within by using the element number.

Arrays are useful for storing and checking a range of items. For example you might want to check the status of a grid of items or access a piece of information from one part of a table.

Note: In keeping with standard programming practice arrays start counting from zero. Loop counters also start at zero. If you want to start counting your array elements at 1 you will need to make the array 1 larger than it should be and ensure that loops you use with the array also start at 1.

Creating Arrays

You create an array by declaring it in a script object or using a script action. You can choose to have an unlimited array or one with a set number of elements.

To declare an array type the following into a script.

```
var arrayName = new Array()
```

or

```
var arrayName = new Array( number )
```

var – the keyword **var** is used to declare and initialise a new variable.

arrayName – the name you will give to the Array.

new Array() – the keyword **new** is used to create a new object and **Array** is the object type you are creating.

() – the brackets contain the number of elements in the array. If the number of elements is omitted, the number of elements can be expanded as required.

When you create an Array, you are said to be declaring the array. In this example a new array is created using the variable `day` and it will contain seven elements

```
var day = new Array(7)
```

You are not required to put a number in the Array keyword. There are many times when you won't know how many elements there will be in an array, so you can always just create an empty array and populate it, as and when you need to. For example:

```
var day = new Array()
```

Using an Array

Once you have declared an array you can then use it, firstly by filling it with information and then by accessing it. To access the array use the following in a script.

```
arrayName[ elementNumber ]
```

arrayName – the name you gave to the Array.

elementNumber – the number of the element within the array. The first element is number 0, the second is 1 and so on.

[] – the elementNumber must be surrounded by square brackets.

Note: You can use the value of an element in a statement.

Populating the array

When you first declare an array you can populate each element or you can populate an array at any time. The array name is suffixed with a square bracket containing the element number. In practice with standard programming, arrays start with 0 as the first element. For example:

```
day[0] = "Sunday"  
day[1] = "Monday"  
day[2] = "Tuesday"  
day[3] = "Wednesday"  
day[4] = "Thursday"  
day[5] = "Friday"  
day[6] = "Saturday"
```

Populate an array with a loop

A common method of populating an array is by using a loop. For example:

```
dollars = new Array()  
sterling = new Array()  
sterling[1] = 23  
sterling[2] = 17.50  
sterling[3] = 75  
var exchangeRate = 1.70  
var loopCounter = 1  
while (loopCounter <= 3)  
{  
    dollars[loopCounter] =  
sterling[loopCounter] * exchangeRate  
    loopCounter++  
}
```

In the example above, the array named **sterling** is multiplied by the value of **exchangeRate** and the calculated result stored in an empty array named **dollars**. Notice that the element number can be a variable (as long as the variable is a number). The first time the **while** loop is executed the variable **loopCounter** is **1**, therefore the element **sterling[1]** is multiplied by **exchangeRate** (i.e. 23 multiplied by 1.70) and the calculated result stored in the element **dollars[1]**.

Using an element in a statement

Each element in an array is like a variable, it can contain any of the Data Types, such as a string, a number or a Boolean value. As a result elements are often used in other lines of codes in your program

```
var today = new Array()  
today[0] = "Sunday"  
var firstDay = day[0]  
var checkDay = "Sunday"  
if (checkDay == day[0])  
{  
    sundayImage.Show()  
}
```

The third line in **Example 4** above stores the value of element **day[0]** to a new variable named **firstDay** (i.e. **firstDay** now contains the value "Sunday"). In line 4, the variable **checkDay** is set to the string "Sunday" and the **if** expression checks if the variable **checkDay** and the element

day[0] contain the same data, if they do the **Image** object named **sundayImage** is displayed on the page.

ListBoxes and Tables

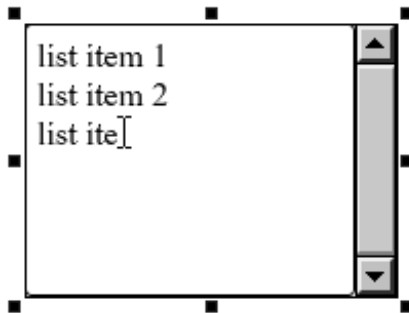
ListBoxes

Professional versions of Opus allow you to create scrolling listboxes of elements which allow users to select from a range of options via one object.

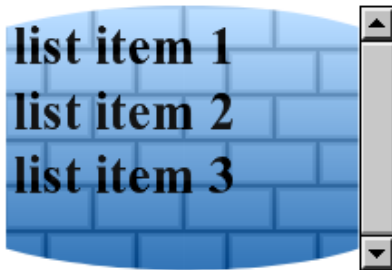
Creating a ListBox

You can create a list box by drawing it onto your page just like any other object.

Once drawn onto the page you can click in the main panel of the listbox to add the elements of the list by typing them each onto a new line.



You can edit its properties to suit your particular requirements, adding a background, images, texture and so on, so that it doesn't even need to look like a standard Windows list box.



Properties specific to the ListBox object can be edited on the ListBox tab. In particular you will need to store the selection the user has made to act upon it using script or programming actions such as **If** or using the **Variable Changed** trigger.

Index of Selected Item

Store index of selected item

ListBox1Selection ▼ New...

Check this box and choose a variable in which to store the selection made by the user as it place in the list (its index). This option is useful if you then want to use the list box in conjunction with an array.

Content of Selected Item

Store selected text

Choice1 ▼ New...

This allows you to store the text which has been selected.

Selection Style

You can also design a style for the way in which the current selection will be displayed in the list box.

Selection Style

Blue Selection ▼ Customise...

A series of presets are provided or you can create your own using the **Customise** button.

Create As Table

Opusvides all the commands you need to create most elements of a program and we have deliberately avoided pre-building these in order to keep the size of the publication down to the minimum and to allow as much versatility as possible. We do not provide a table object as such but this function allows you to build a table quickly and easily from a whole range of different elements.

The Create Table option is part of the QuickBuild menu and effectively allows you to duplicate an object and copy it a specific number of times across and down the page with a specific space between each copy:

If the object already has content the content will be duplicated

This feature is particularly useful when you want to create a table of Text objects, for example, to show rows and columns of data.

The Create Table feature has an extra facility when used with a Text object – you can automatically have data from an Array variable fill-in the rows and columns with the elements of the array. This is a very quick method of creating a table and contents in one go – see Adding Array information with the Create Table feature below for more information.

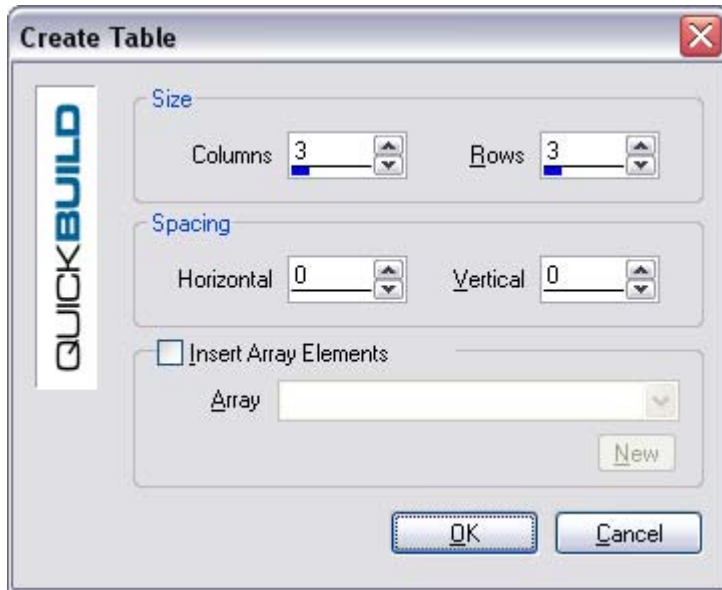
Using the Create Table feature:

1. Select the object you want to duplicate to create the table – a bounding box will surround the object.

If you want all the copies in the table to appear exactly the same, we recommend you make all of the changes in the original copy's Properties dialog before you use the Create Table feature.

2. Select Create Table from the QuickBuild menu – this will open the Create Table dialog box.

Alternatively, right mouse click over the object to open the right-click menu and select Create Table from the QuickBuild sub-menu.



3. Use the Size panel to set the number of copies across the page and down the page you require of the selected object. Use the spin buttons beside the Columns box (i.e. number across the page) and Rows box (i.e. numbers down the page) to set the number of copies. You can set different numbers in each box.

Note: If you only want columns, then set the Rows box to 0 (i.e. zero). Alternatively, if you only want rows, then set the Columns box to 0.

4. Use the Spacing panel to set space that should appear between each object copied. Use the spin buttons beside the Horizontal box (i.e. space between rows) and Vertical box (i.e. space between columns) to set the number of pixels. You can set different numbers in each box.

If you want no spaces between each object, set the box to 0 (i.e. zero) for either or both the Horizontal and Vertical boxes.

5. If the object to be copied is a Text object, the Create Table dialog contains an Insert Array Elements panel. If you want to use this option, it must be filled-in before you do option 5 below – see Adding Array Information with the Create Table feature below for more information.

6. Click on the OK button to save your changes – Opus will automatically create the table for you.

Each object is an exact copy of the original but can be edited and changed, just like any other object. The new objects are named after the original copy, for example, if the object copied is named Box, the first copy is Box (2), the second Box (3), and so on.

You can use the Rename option in the QuickBuild menu to change the name and number sequence of the objects – see Rename for more information.

Alternatively, you can use the Rename from Contents option in the QuickBuild menu to change the name of objects based on their contents – see Rename from Contents for more information.

Adding Array information with the Create Table feature:

1. Do steps 1 to 4 of Using the Create Table feature above first – make sure you have selected a Text object.

2. Tick the Insert Array Elements panel checkbox if you want the value of an element in an Array variable to be added to one of the Text objects created by the table. If you do not tick this box, then the newly created Text objects will remain blank (or have the same contents as the original if it contained text).

3. Click on the down arrow next to the Array box to list all of the Array variables in the current publication.

If the Array box is empty, you have not yet created an Array variable in your publication. If you want to use this option, you must have created at least one array (see point 4 below).

4. If you have not yet created the Array variable, click on the New button – this will open the New Variable dialog box in which you can create a new Array variable.

5. Click on the OK button to save your changes – Opus will automatically create the table for you.

6. The Array elements are added to the table in the following way. The top left Text object is the original object selected to create the table, this will contain the element at Index position [0]; the first copy in the table is given Index position [1], the second copy Index position [2], and

so on, until each copy has been given the next Index position in the sequence.

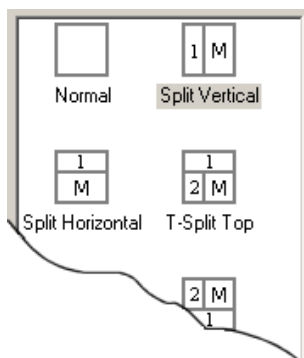
To set the contents of each element in an array, you must use the Set Variable Array action or write an expression in the Insert Variable or Expression option on the Text objects right-click menu.

Layout

The Pro versions of Opus allow you to layout your publication as a combination of pages. This allows you, for example, to have a menu panel which remains consistent and a content panel which displays different pages depending on the selection on the menu panel. Thus your user need not move backwards and forward between a separate menu page but instead has immediate access to the menu.

The **Layout** tab is on the **Publication Properties** dialog and allows you to create a publication with multiple panels open at one time. A panel is simply another name for a window. Each panel is active and can display different pages, each of which can contain any type of object, such as images, buttons, text, slideshows and more.

By default when you open a new publication the layout is set to 'Normal', which means the only available panel is the 'Main Publication Window'. However, you can select an alternative to the 'Normal' layout by selecting one of the icons from the list in the left-hand column of the Layout tab.



The Panel list in the right-hand column of the Layout tab will display the names of the Panels for the Layout Type selected in the left-hand column. In other words, if you leave the Layout type on the Normal selection, the Panel list will display 'Main Publication Window'; if you select Split Vertical as the Layout Type then 'Panel 1' is added underneath 'Main Publication Window'.

Panel Properties

Each panel listed in the Panel list, has properties that must be individually set. Select the panel name in the Panel list and then decide on the width and height of the panel...



The width and height of each can be set to either an Absolute, Percentage or Relative value.

An **Absolute** value is set to a number of pixels, for example, the panel could be set to 100 pixels wide and 100 pixels high.

A **Percentage** value is set as a Percentage of the screen size, where 100% is the full vertical width or height of the screen. If the panel is using a window and not the full screen, then 100% is the full size of the window.

A **Relative** value is a little more complicated to explain. A relative value divides up the space left after the absolute and percentage amounts are taken out. The relative value number is a proportion of the screen or window that the selected panel gets. For example, consider a three panel layout with a screen that is 100 pixels wide. If the widths of the panels were set so that the first panel was set as a percentage of 30% and the second and third panels were set to a Relative value of 1, then the panel width in pixels would be 30 for the first panel, 35 for the second panel and 35 for the third panel. In other words, the first panel is 30 pixels of the total width of 100 and the other two panels get an equal share of the remaining 70 pixels.

Now if the widths of the same three panels were set to an Absolute value of 25 pixels for the first panel and a Relative value of 2 for the second panel and a Relative value of 1 for the third panel, then the panel width in pixels would be 25 for the first panel, 50 for the second panel and 25 for the third panel. The first panel gets its 25 pixels and the other two panels share the remaining 75 pixels, but with the second panel getting twice as much as the third.

Once you have set the width and height for a panel you can also add vertical and horizontal dividers...

Vertical Divider Horizontal Divider

Tick the vertical or horizontal divider option if you want a line to appear between the different panels when the publication is run. If this option is not ticked, no divider will appear between the panels on screen.

Finally, if you want users to be able to resize the panel vertically or horizontally...

Vertical Resize Horizontal Divider

Tick the vertical or horizontal resize option if you want the user to be able to resize the panel to their own size when the publication is run. If this option is not ticked, the size is fixed.

Setting Panels to Chapters

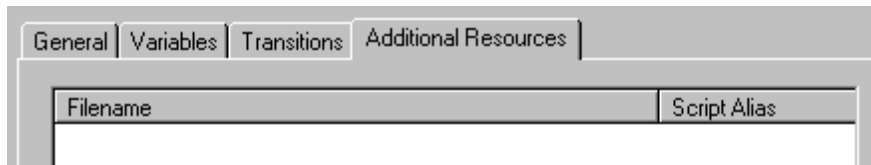
Only one Panel type can be used within each Chapter, that means all the pages within the same Chapter will always use the same Panel.

Publication Properties

In addition to the Layout tab described previously Professional versions of Opus also provide two other tabs on the Publication Properties dialog which are worthy of note. (Other options are also available on other tabs – two are mentioned below and the others can be reviewed via the Help file).

Additional Resources

This section of the Properties dialog allows you to add additional resources to your publication that will be referred to in OpusScript. Each new resource can be given an ‘alias’, which acts as a shorthand method of referring to the full pathname for the resource. Resources can be any type of file, such as a video clip, an audio sound or a sound.



The **Additional Resources** tab appears in both the **Publications** Properties dialog and the **Page** Properties dialog. The resources added in the Publication properties dialog can be used in an OpusScript on any page in your publication, while resources added to a page can only be used in OpusScripts on that page.

There are three buttons available in the **Additional Resources** tab:



To add resources to the Additional Resources tab, click on the Add button, which will open a dialog box and do the following:

1: Your first choice is to decide whether you want to add a single resource or a range of resources.



2: Click on the **Add Single File** radio button to add a single resource. Click on the button to the right of the field to browse your system for your resource.



3: Alternatively, you can decide to add more than one resource by clicking on the **Add Directory** radio button. This option has a number of other settings that allow you to filter the type of resources that you want to add.

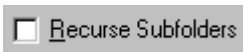


Click on the **All Files** radio button to add all the files contained in the selected directory.



Or click on the **Wildcard** option to only include resources that match your entry in the field. For example, to include only jpg files from the directory, you would enter in the field '*.jpg'; to include all files beginning with 'voice' (whether they are jpg, wav, avi or any other file type) you would enter 'voice*.*'

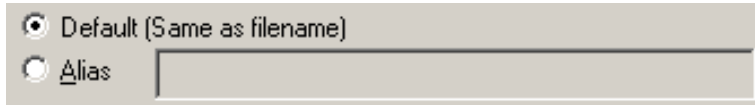
The final option available in **Add Directory** is **Recurse Subfolders**, this option is available whether you decided to include all files or enter a wildcard.



Check this box if you want to add all resources that appear in sub-folders of the folder you entered in **Add Directory**. Leave the check box blank if you do not want to add resources in the directory's sub-folders.

4: The final section in the dialog box is **Script Alias**, in which you can provide an 'alias' name for resources. Any alias name you enter for a resource is displayed in the **Additional Resources** tab in the **Script**

Alias column beside the **Filename** column. There are two radio buttons available in the Script Alias



The **Default** radio button will keep the full pathname of the resource e.g. c:\sounds\splash.wav. When referring to this file in OpusScript you would need to enter the full pathname e.g. PlaySound(“c:\sounds\splash.wav”).

Alternatively, you can have one alias name that is prefixed to the added resources. Click on the **Alias** radio button to enter an Alias name e.g. “splash”. The alias name can be used to refer to the file in an OpusScript e.g. PlaySound(“splash”).

Note: If you selected **Add Directory** and you want to enter an alias name, the alias is then prefixed to the filename of the resources. For example, if a directory contained six files named ‘pic1.jpg’ to ‘pic6.jpg’ and you entered the alias ‘sequence’, the result would be an alias name of ‘sequence\pic1.jpg’; ‘sequence\pic2.jpg’; and so on.

5: Click on the **OK** button to add the resources to the **Additional Resources** tab.

Removing Resources

To remove an added resource from the Additional Resources tab:

1: Open the **Additional Resources** tab in the **Publication Properties** or **Page Properties** dialog in which the resource was added.

2: Select the resource in the Filename column.

3: Click on the **Remove** button and then click the **Apply** button to save your changes.

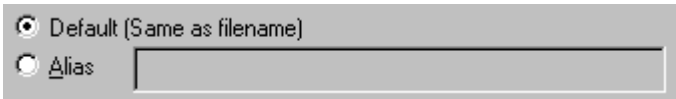
Editing Resources

Resources that have been added to a page or publication can be edited. You may edit a resource to give it an alias name or to rename it to a new alias name or default back to its original path name. To edit an additional resource:

1: Open the **Additional Resources** tab in the **Publication Properties** or **Page Properties** dialog in which the resource was added.

2: Select the resource in the Filename column.

3: Click on the **Edit** button to open a new dialog box. The new dialog box allows you to create or remove a Script Alias, as described above.

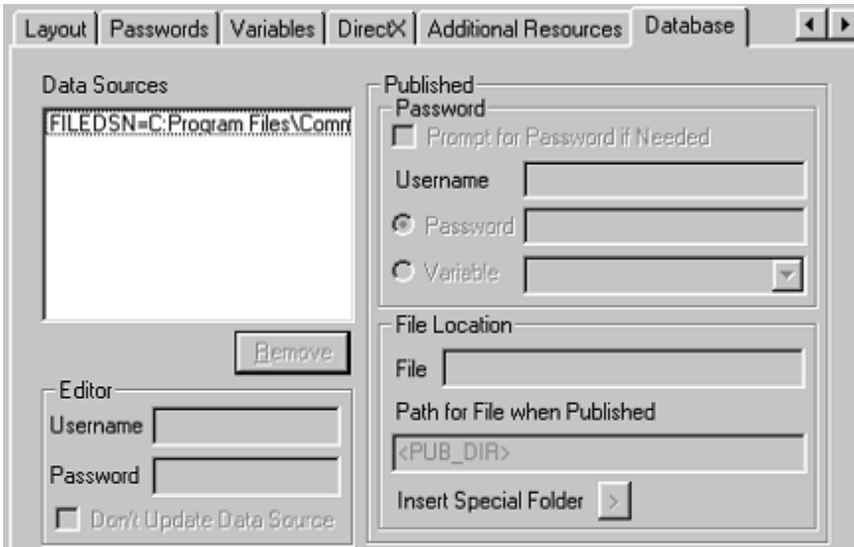


If the selected resource currently has an alias name, it will be displayed in the Alias field; the alias name can be changed to a new name.

4: Once you have made your changes, click on the **OK** button to close the new dialog and click on the **Apply** button to save your settings.

Database

This tab allows you to set up general information about the location and security of any databases used in the current publication.



Data Sources

The Data Sources window lists all DSN connections to databases currently attached to this publication. To remove a database connection in your publication.

1. Highlight the appropriate Data Source. The **Remove** button will be enabled.
2. Click the **Remove** button to remove the database connection in this publication. Please note, this option will not automatically ask for confirmation prior to deleting the database connection.
3. Click on the **Apply** button to save your settings.

Editor

Databases can be password protected, ensuring secure access to data for authorised users only. Password protected databases can be accessed in Opus when you enter an authorised username and password. Whenever you attempt to create or edit a database Action (e.g. the 'Database Query' or 'Next Record' Actions) you will be prompted to enter your username and password.

This option allows you to store the username and password within Opus so that you are not prompted to enter the details every time you create or edit a database Action. To store the username and password for a Data Source:

1. Highlight the appropriate Data Source. The Username and Password fields will be enabled.
2. Enter your Username and Password in the appropriate fields.
3. Click the **Apply** button to save your settings.

Note: Setting a username and password in the Editor option will only affect access to a database at development time and not when the publication is published (see Published below).

'Don't Update Data Source' Check Box

The Data Source's currently connected to a publication are listed in the Data Source window above this check box. Each Data Source has properties that are checked when the publication is first opened e.g. to

check if the name of the database has changed or the type of ODBC driver used is different.

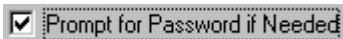
Placing a tick in the check box will ensure that the properties for a Data Source are NOT checked when the publication is first opened. This option is very useful for two reasons: one, when you want to send the publication to another user (because they do not need the Data Source on their computer); two, when another user has a Data Source with the same name on their computer.

Note: Checking this option does not affect the database Action Triggers i.e. you can still update current records and insert new records when the publication is published.

Published

This option allows you to store the username and password within Opus so that the end-user is not prompted to enter their authorisation details when they run the publication.

Tick check box to ensure the end-user has to verify authorisation:



Clear check box to ensure the end-user does not have to verify authorisation:

Note: More than one username and password may be valid for each Data Source. To enter the username and password:

1. Enter the user's name or a variable name in the Username field.

When entering a variable name, ensure it is surrounded by chevrons e.g. <UserName>.

2. Select either the **Password** or **Variable** radio button. Use the Password field when the authorisation password is the same for every authorised user. Use the Variable field when more than one authorisation password is used to access the database. Select the variable name from the drop down list in the 'Variable' field.

3. Click on the 'Apply' button to save your settings.

File Location

The File Location option can be used to set where the database file will be stored when the publication is published. The 'File' field contains the current path of the database file. Please note, if this field is greyed-out then this Data Source does not use a file but uses instead a proprietary database application, such as SQL Server or mySQL.

Opus provides a list of special folders where the Data Source file may reside when the publication is published:



Each of these folders have been given a system variable name within Opus. If you select one of these options the system variable name will be displayed in the 'Path for file when Published' field.

Image Background

In the **Publication Properties** dialog you can add an image in the Image tab for full screen publications. The image will replace the surround colour set in the Options tab of the **Publication Properties**. If your publication has a page size of 800x600 but is displayed on a monitor set to a larger screen size (e.g. 1024x768), the image will appear behind the page but fill the screen. This can be a stunning effect and will give any presentation an extra look of professionalism.

Minimise Options

When you run Opus Presenter in **Extended** mode, you can access the **Publication Properties** dialog. The **Windows** tab in this dialog contains two new options:

1. Use the **Run Minimised** option if you want the publication to initially open in the Windows desktop Task Bar rather than displayed full screen or in a window. This is useful if you are

creating a publication like a calendar, diary or alarm for occasional use.

2. Use the **Minimise to System Tray** option if you want to be able to minimise the publication in the Windows desktop System Tray when the user minimises the publication.

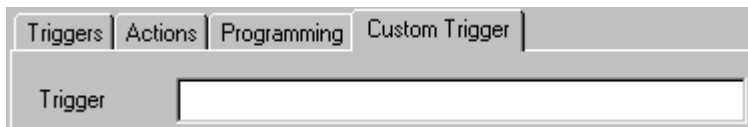
This is often used with the **Run Minimised** option (point 1 above) to create background programs. If the **Run Minimised** option is applied then the publication is not shown on the page, however, it is processing as soon as it is opened. This means your publication is working even though it is not displayed making this option very useful for background processing publications.

Custom Triggers

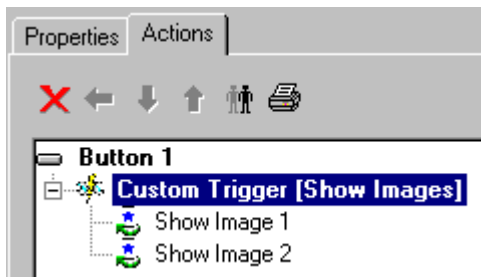
You can have an action or a series of actions occur when another action called a **Custom Action** is activated. Actions normally only happen when a trigger is activated but the Custom Action is different, its sole purpose is to activate a Custom Trigger.

This effectively means that you can have one action that sets off a series of other actions, it also means that the same series of actions can be activated by different triggers without you having to duplicate the actions in each trigger, which is particularly useful if you change the list of actions because you don't have to make the changes more than once. For example, you could have the same sequence of events occur when a user clicks a button object, moves over an image object or enters a correct answer in a text object by simply adding the Custom Action to each object's trigger which will then activate the Custom Trigger.

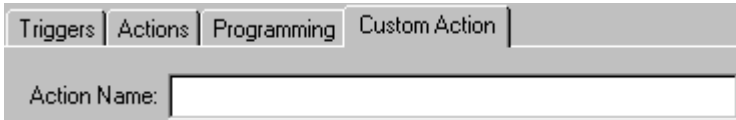
When you select the Custom Trigger a **Custom Trigger** tab will appear after the Programming tab in the Actions dialog.



In the Trigger field, enter the name you want to give to the custom trigger. You can enter any name and it can include spaces, for example, you could name the trigger 'Show Images'. As with any trigger, you must then add the actions you want to happen as a result of the trigger being activated, for example to show 'Image 1' and 'Image 2'.



Next use a Custom Action to activate the Custom Trigger. The Custom Action is in the **Programming** tab of the **Actions** dialog and it can be added to any object on a page. The Custom Action has one entry:



The image shows a dialog box with four tabs: 'Triggers', 'Actions', 'Programming', and 'Custom Action'. The 'Custom Action' tab is active. Below the tabs is a text input field with the label 'Action Name:'.

Enter the same name you entered in the Custom Trigger in the 'Action Name' field and click the **Apply** button to save your settings.

What will happen is this: When the trigger containing the Custom Action is activated, the Custom Action will search for a Custom Trigger in the current object, if it doesn't find one it will search each object on the page for a Custom Trigger. If it finds one it checks if the Action Name in the Custom Action matches the Trigger name in the Custom Trigger. If it does, it then runs the Custom Trigger actions.

Activating a Custom Trigger using OpusScript

A Custom Trigger can also be activated within a Script Action or Script Object using the `TriggerComponentEvent` script function (see the *Event Script Actions* section for more information). This function can be added to any Script Object or Script Action (see the *Script Objects* and *Script Actions* sections for more information) and it requires one parameter, which is the name you have given to the Custom Trigger. In the illustration above the Custom Trigger 'Show Images' has been added to the Object named 'Button 1'. To activate the trigger you would enter the following line in a Script Object or Script Action:

```
Button_1.TriggerComponentEvent("Show Images")
```

What happens is that whenever this line is executed in your script, the system will look for an object named 'Button 1' on the page, it will then look for a Custom Trigger named "Show Images" in the object and finally, if it finds the Custom Trigger it will then perform the actions listed underneath the trigger.

Using Parameters

In addition to passing parameters when launching other files it can be useful to accept parameters when opening your publication. The most common use of this is for going to a particular page. You may, for example, have another log-in publication which manages a series of tests or information publications and directs particular users to particular points.

Professional versions of Opus allow you accept parameters when your publication is launched, allowing you to be provided with particular information at start up. Parameters are additional information provided to a program when it is launched and in this case they are preceded by a forward slash character.

How to use Parameters

1. Opus checks to see if parameters are passed whenever a standalone publication is run. Parameters can be passed when one Opus publication opens another Opus publication using the Launch File action or when any other program launches the Opus publication.

2. If a Launch File action contains parameters, the following process will happen. Opus will open the new publication and create a `COMMAND_PARAM` variable for each parameter supplied. For example, if the program launches the publication with the following parameters **/Page2 /On** the following variables will be created in the new publication:

`COMMAND_PARAM_1` will contain the value Page2

`COMMAND_PARAM_2` will contain the value On

3. Your new publication now has two new `COMMAND_PARAM` variables, which you can use as and when you like – Please note point 4 below!

You must remember the order in which you passed the parameters so you know which one to use in your new publication. For example, if the first parameter you sent contains a page name and the second a value to turn music on or off, you must remember that `COMMAND_PARAM_1` is used when you want to go to a specific page.

4. In the new publication (i.e. the one using the parameters), you **MUST** create the new `COMMAND_PARAM` variables when you create your publication. This is important! The `COMMAND_PARAM` variables do not already exist in your publication, that is, you cannot pick them off the list of variables available.

The reason the `COMMAND_PARAM` variables do not exist is because we do not know how many parameters you may pass from one publication to another. The variables are created ‘dynamically’, which simply means that when you use the Launch File action in one publication to open a new publication, Opus counts the number of parameters you are passing and then creates the exact same number of `COMMAND_PARAM` variables in the new publication.

5. To add `COMMAND_PARAM` variables to your new publication, simply create one variable for each parameter. For example, if you are sending three parameters to your new publication, then create three variables with the following names:

`COMMAND_PARAM_1`

`COMMAND_PARAM_2`

`COMMAND_PARAM_3`

To create a variable in Opus – see *Creating Variables* in the Help file for more information.

The `COMMAND_PARAM` variables do not have an initial value because that information will come from the Launch File action in the publication that opens your new publication.

6. Often, parameters are passed because you want to change the behaviour of your publication when it is first opened. For example, publications always open on the first page by default, but by passing a

parameter, you can make a different page appear when the publication is first opened.

Although parameters are most commonly used when you first enter a publication, they can be used at any time in your publication. For example, you could have a parameter that specifies whether a voiceover should be On or Off. Any page that contains a voiceover could first check the parameter to see if it should play or mute the voiceover.

7. To use a parameter when the new publication is first opened, use the Pre-Show trigger on the first page of your publication. For example, if a parameter passed to the new publication specifies the first page to display in the publication, you must use a Pre-Show trigger to ensure the first page does not appear briefly before you go to the specified page – see Example 1 below for more information.

8. Use the Select and Case actions when the parameter contains one option from a variety of options. For example, if the first parameter contains a page name, do the following. Set up a Select action that uses the PARAM_COUNT_1 variable; create a Case action for each page name the parameter can contain; add a Go to Page action to each Case action that will open the page you want to open.

Remember to add the Select and Case action to a Pre-Show trigger on your first page of the new publication. This will ensure that the first page the user sees in the new publication is the specified page and not the first page in the publication.

9. Use the IF action when the parameter contains one of two possible options. For example, if the second parameter contains a value that turns a voiceover On or Off, do the following. On the page or pages in your publication that this applies: add an If action that uses the variable COMMAND_PARAM_2; use a Set Variable action to test if the parameter contains the value On; add a Play Sound action to the If action to play the voiceover. In this example, the voiceover will only play when the second parameter is set to On, if it is set to Off (or anything else), the voiceover will not play.

10. Use the Insert Variable or Expression option in a Text object if you want to display the contents of a parameter on the page. For example, if the third parameter contains the value of the variable UserName, do the following. Click inside the text box you want to add

the user name to; right-click in the text box to open the right-click menu and select the Insert Variable or Expression option; in the Insert Variable or Expression dialog add the variable `COMMAND_PARAM_3`. In this example, the user name will be displayed in the new publication. See [Displaying Variables on a Page](#) for more information.

Notice you enter `COMMAND_PARAM_3` and not the variable `UserName`. This is because the Launch File action can pass the contents of a variable as a parameter, however, it is the value contained in the variable and not the variable name that is sent. Therefore, if the third parameter was set up to use the contents of the variable `UserName`, and the name was John, then `COMMAND_PARAM_3` will contain the value John.

A worked example is provided in the Help file.

Additional Launch Actions

This subsection of the Actions dialog provides actions to launch the publication search dialog, a web page or another program. Opus provides two advanced actions...

Post Web Data



This option allows you to send and/or receive data from a remote Server. You could send Quiz score results from your publication to a Server so all results from tests could be centrally stored - or you could receive data from a Server into your publication, such as the latest price list, which means you don't have to re-publish a new version of your publication every time prices are changed. Information is sent or received using variables.

1: Enter the URL (Uniform Resource Locator) that you want to send data to or receive data from.

URL:

The URL is the address on a remote server that the information will be sent to or received from. Normally, this will be to a perl script or some other program that will process the data. These scripts are normally saved in a subfolder, which is often named cgi-bin. The URL will look similar to this:

`http://www.digitalworkshop.com/cgi-bin/results.pl`

2: There are two tabs named **Source** and **Destination**. You must fill in both tabs, however, the options available on both tabs are exactly the same.

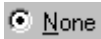
Source | Destination

The only difference between the tabs is that:

Source is used to send data from the Opus publication to the remote server.

Destination is used to receive data from a remote server into an Opus publication.

Both tabs contain three radio buttons:



The first radio button is **None**. This option will do nothing! In other words, if you don't want to send information to a remote server then make sure None is selected in the Source tab; if you don't want to receive data from a remote server, then select None in the Destination tab.

Note: If None is selected in both the Source and Destination tab – then nothing will happen... at all. Often you will want to send or receive data but not do both at the same time, so one of the tabs will probably be set to None.

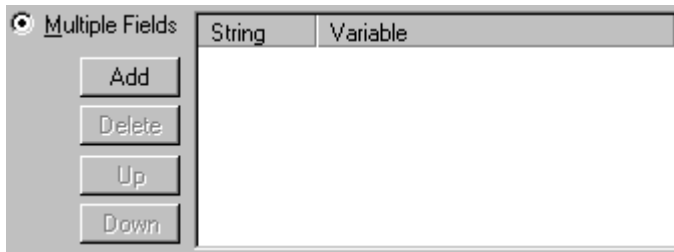


The second radio button is **Variable**. Use this option when you want to send one single bit of information to a remote server or receive a single bit of information from a server. Click on the down arrow to the right of the Variable field to list all of the variables currently in your publication.

Sending information to a server: Select the variable that contains the information you want to send. The variable can be any of the system variables, such as SCORE_TOTAL or a variable you created.

Receiving information from a server: Select the variable in which the received data should be stored. The variable can also be a system or user-defined variable.

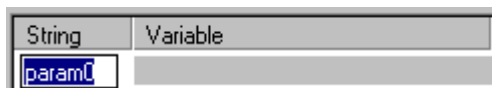
Note: The script that processes the data on the remote server works by extracting name-value pairs. In other words, it looks for a specific name and then the value it contains. When sending or receiving a single value to a remote server, the variable name is Param0 and the value is the contents of the variable you entered in variable. Therefore, the perl script should be looking for the name Param0.



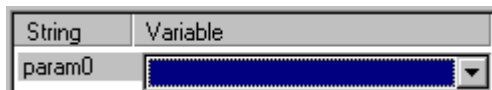
The final radio button is **Multiple Fields**. Use this option when you want to send more than one bit of information to a remote server, or receive more than one bit of information from a server.

Click on the **Add** button to add a new entry. You have to enter two bits of information for each field you want to send or receive:

The first is **String**, which is the name of the field the perl script will search for in a name-value pair. By default, the first time you click the Add button, the string is called param0, the second time param1, and so on. You can change the default name by clicking on the string name once, which will then allow you to enter your own name for the field.



The second bit of information you must add is **Variable**. This is a variable name which will either contain the value sent back from a remote server or contains the value that you want to send to the server. To add the variable name, click in the space beside the String name once, this will show a drop-down list box from which you can select your variable name.



You can continue adding new entries by simply clicking the **Add** button again.

Once you have added one entry in the list, the **Delete** button will become active. To delete an entry, select the entry from the list and click on the **Delete** button.

When you have entered more than one entry in the list, the **Up** and **Down** buttons will become active. You can re-order the position of an

entry on the list by selecting the entry and then clicking on the **Up** and **Down** button as required.

3: Finally, once you have made all your changes, click on the **Apply** button to save your settings.

Clock Actions

This subsection of the Actions dialog is new to Opus and allows you to create and control a clock that can be displayed on a page in a variable. The clock always start at zero and counts up in seconds. You can display the seconds, minutes and hours in the clock.

When a clock is created it can continue to run while other pages are displayed in the publication. The clock can even be created on a Master Page and appear on each page that uses the Master.

Start Clock



This option is used both to create the clock and to start the clock. When you select this action, a **Start Clock** tab will appear after the Programming tab in the Actions dialog. The first option in the tab allows you to select the name of a previously created clock from the drop down list to the right of the Clock field, or to Add a new clock.



Click on the **Add** button to create a new clock. The name can contain alphabetic or numeric characters and can contain spaces e.g. these names are all valid: 'My Clock', 'Countdown 1', '1 countdown'.

The Variable field contains the name of the variable that will contain the Clock time. The variable must be a publication variable and not a page variable.



Click on the **New** button to create a new variable. Alternatively, select the name of a variable from the drop down list to the right of the Variable field. This is the variable that can be displayed on screen to show the current Clock.

The Format option allows you to select the way the clock should appear on screen.

A screenshot of a software interface showing a dropdown menu labeled 'Format'. The menu is currently empty, and a small downward-pointing arrow is visible on the right side of the dropdown box.

The format options are:

[hh] – equals hours, shown as two digits e.g. 01 or 23

[mm] – equals minutes, shown as two digits e.g. 09 or 59

[ss] – equals seconds, shown as two digits e.g. 00 or 59

Other alphanumeric characters can be placed in the Format field, for example, you may want to separate the hours, minutes and seconds with a semi-colon, like so:

[hh]:[mm]:[ss] e.g. 09:23:54 (9 hours 23 minutes and 54 seconds)

However, other formats can be created, for example this is also a valid format:

Minutes: [mm] and Seconds: [ss] e.g. Minutes: 23 and Seconds: 54

Stop Clock



This option will stop the specified Clock, the current time will remain displayed on the screen. When you select this action a **Stop Clock** tab will appear after the **Programming** tab in the **Actions** dialog.

This action has one option, which is the name of the Clock that you want to stop. Select the name of the clock from the drop-down list that appears to the right of the Clock field.

A screenshot of a software interface showing a dropdown menu labeled 'Clock'. The menu contains the text 'Clock1' and a small downward-pointing arrow is visible on the right side of the dropdown box.

Once a clock has been stopped it can only be re-started using the Start Clock action, this will reset the clock back to zero. You must use the Pause Clock action instead of the Stop Clock action if you want to continue the clock from its last position, the clock must then be re-started using the Continue Clock action.

Pause Clock



This option will pause a specified Clock, the current time will remain displayed on the screen and the Clock can be resumed with the ‘Continue Clock’ action. When you select this action, a **Pause Clock** tab will appear after the **Programming** tab in the **Actions** dialog.

This action has one option, which is the name of the Clock that you want to pause. Select the name of the clock from the drop-down list that appears to the right of the Clock field.

Once a clock has been paused it can resume counting from its current time, when a ‘Continue Clock’ action is activated. If a ‘Start Clock’ action is triggered, the current time is lost and the clock is reset to zero.

Continue Clock



This option will resume the counting of a specified Clock only if the clock has previously been stopped with a **Pause Clock** action. When you select this action, a **Continue Clock** tab will appear after the Programming tab in the Actions dialog.

This action has one option, which is the name of the Clock that you want to resume counting. Select the name of the clock from the drop-down list that appears to the right of the Clock field.

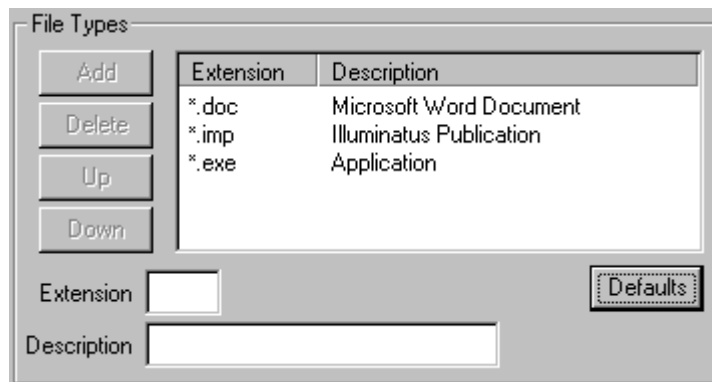
File Management Actions

This subsection of the Actions dialog allows the user to open, copy or delete files while a publication is running. Each of the options below use the Window dialog boxes to carry out the function.

File Browse



This option allows you to set up a list of file type Extensions (such as exe files, imp files, and so on), that a user can search for on their system using the familiar Windows Browser dialog. When you select this option, a 'File Browse' tab will appear after the Programming tab in the Actions dialog.



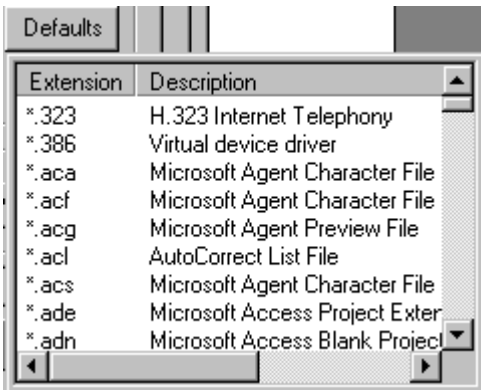
Extension	Description
*.doc	Microsoft Word Document
*.imp	Illuminatus Publication
*.exe	Application

There are two methods of entering an Extension. The first is to enter the Extension name in the Extension field and the description in the Description field. Once you have entered the Extension and Description, the Add button will become active, click the Add button to add it to the table. Each new Extension added to the table will automatically appear as the bottom entry in the table. You can move the position of the Extensions in the table by highlighting the Extension name in the table, which will activate the Delete, Up and Down buttons.

The Delete button will remove the Extension name from the table. The Up button will move the Extension up one place in the list and the Down button will move the Extension down one place in the list. The order of the Extensions in the table is significant because when the Browser dialog is displayed the 'Files of type' field will list the Extensions the user can search for in the same order as they appear in the table.

Note: the Extension name should not be prefixed by a full stop e.g. '.exe' – simply enter 'exe'. The system will automatically prefix the Extension with a full stop when the Extension and Description are entered in the table.

The alternative method of adding an Extension to the table is by clicking the Defaults button.



The defaults button will display a list of all the common Extension and Description names found on a Windows system. Find your required Extension and double-click on the entry to add it to the table. As with the previous method, the Extension is added to the bottom of the table and you can re-order its position using the Up and Down buttons or delete it from the list by clicking the Remove button.



The final section of the File Browse tab allows you to store the path name of the file the user selected from the Browser dialog into a variable, which can then be used by the Copy File action and Delete File

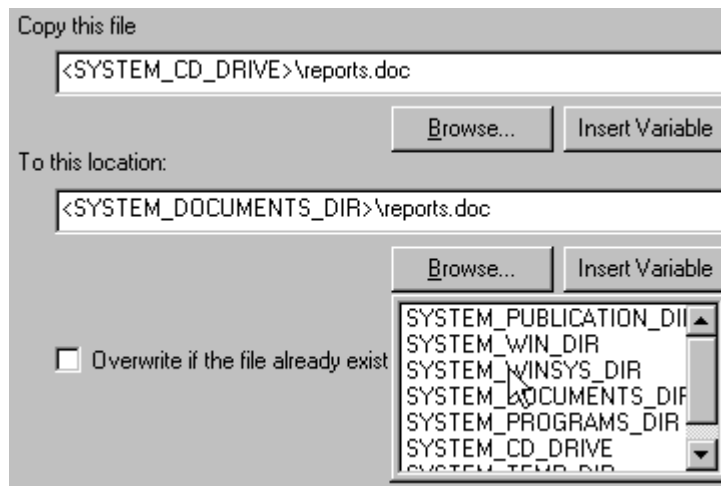
action. A list of previously created variables is available from the drop-down list box that appears to the right of the variable field. Alternatively, you can click on the New Variable button to create a new variable.

Copy File



This option is very useful for copying files from one place to another, especially for copying files from a CD-ROM onto the computers hard disk. For example, this option could be activated by a Left Mouse Click trigger on a Button object named ‘Download Additional Documents’; when the user clicks the button the additional word documents are transferred from the publication CD-ROM onto the users hard disk.

When you select this option, a **Copy File** tab will appear after the Programming tab in the Actions dialog.



In the **Copy this file** field, type in the pathname of the file you want to copy or select the ‘Browse’ button to locate the directory on your system. Alternatively, you can use the Insert Variable button, which lists a number of preset variables that specify particular system locations. In the example above, the preset variable <SYSTEM_CD_DRIVE> has been used, this will automatically search for a CD-Rom drive on the

users system irrespective of the actual letter the drive uses. The name of the actual document is added to the end of the variable name, prefixed with a backslash (see graphic above).

In the **To this location field**, type in the pathname where this file should be copied. As with the previous field, you can type in a pathname, or use the Browse or Insert Variable buttons to enter the path and filename. You must enter the name of the file with the appropriate extension file type e.g. .doc, .exe., .imp etc. The copied file can be given a different file name if required.

Tick the checkbox **Overwrite if the file already exists** if you want to replace an existing file in the destination directory that has the same name as the new copied file. Alternatively, do not tick this checkbox and the copied file will not be placed in the directory.

Delete File



This options allows you to delete a file from the system, which may be useful if you temporarily create files when a publication is running and you want to delete them when the user leaves the publication. When you select this option, a 'Delete File' tab will appear after the Programming tab in the Actions dialog.



The image shows a dialog box titled "Delete this file". It contains a text input field for entering a file path. Below the input field are two buttons: "Browse..." and "Insert Variable".

In the 'Delete this file' field, type in the pathname of the file you want to delete or select the 'Browse' button to locate the directory and file on your system. Alternatively, you can use the Insert Variable button, which lists a number of preset variables that specify particular system locations. For example, the preset variable <SYSTEM_PUBLICATION_DIR> will search for the file in the same directory in which the publication was saved on your system. The name of the actual document is added to the end of the variable name, prefixed

with a backslash, for example to delete the ‘functions.doc’ file from the system publication directory, the field would contain the following entry:

<SYSTEM_PUBLICATION_DIR>\functions.doc

The final option allows you to request confirmation that you want to delete the file prior to deleting it.

Display confirmation dialog

Tick this option if you want a confirmation dialog box to appear prior to deletion. Do not tick this box if you want the deletion to happen automatically without confirmation.

Conditional Actions

In the Programming tab of the Actions dialog there are a number of actions that allow you to set an action to occur only if a certain condition is met. Opus provides two new conditional actions that work together...

Select



Opus now provides another decision making feature known as the Select and Case actions. The Select action looks at the value contained within a variable or the value of an expression and checks it against the list of Case actions you have created to see if the value in one of the Case actions is the same as the value in the Select action. If the Select action finds a match in one of the Case actions, it runs the actions listed underneath that Case action and ignores all the other Case actions. If none of the Case action values match the Select action value, a default set of actions will automatically be run instead. This is a very useful tool when you want a different set of actions to occur depending on the value contained within a variable.

When you select the Select action, a 'Select' tab will appear after the Programming tab in the Actions dialog. There are two radio button options, Variable and Expression – only one of the radio buttons can be used for each Select action and the default selection is Variable. The first radio button is Variable.

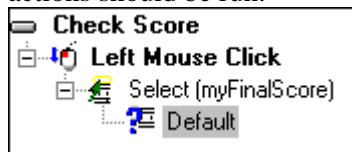


Select the name of a variable from the drop-down list box that appears to the left of the Variable button or create a new variable using the 'Create New Variable...' button. The value contained within this variable is used to check which of the Case actions should be run. As an example, we will assume the variable name entered in the Variable field is 'myFinalScore' and it contains the value 75.

The second radio button is the Expression field. An expression is any valid OpusScript expression (see OpusScript section for more information). Use this option when the value you want to use in the Select action is contained in an OpusScript variable or if you want to combine an OpusScript variable with a page variable. The following line of code is a valid Expression:

```
e.g. var myLastScore = 30
      myLastScore + thisScore
```

In this example, the variable ‘myLastScore’ is an OpusScript variable and ‘thisScore’ is a page variable. Both the values are added together and the final value is the value that is used to check which of the Case actions should be run.



Whenever a Select action is used, a Default action will appear underneath the Select action in the left-hand column of the Actions dialog. The Default action is only activated when none of the values in the list of Case actions match the value in the Select action. A series of actions can be added underneath the Default action by simply clicking and dragging the required action from one of the Action tabs in the right-hand column of the Actions dialog and placing it over the Default action in the left-hand column (see the illustration in the Case action below).

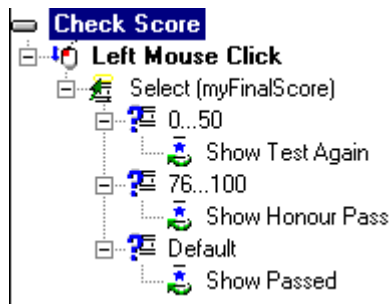
Note: At least one Case action should be included within the Select action.

Case



The Case action is used to contain a value that is checked against the value held in a Select action. If the value in the Select action matches the value in the Case action, the actions listed underneath the Case action are run. This option can only be used when a Select action has already been selected for a trigger.

To place the Case action inside a Script Action, click and drag the Case action icon over the top of the Select action icon in the left-hand column of the Actions dialog and release the mouse button. In the example below, a Select action has been set on a Left Mouse Click trigger for the 'Check Score' button object and it contains two Case actions and a Default action.



When you select a Case action, a 'Case' tab will appear after the Programming tab in the Actions dialog. The tab contains two radio button options: Value and Range – only one of the options can be used for each Case action.



Use the Value option when you know the specific value that you want to match against the Select action's value. To use the example previously used in the Select action description, if you want to check if the variable 'myFinalScore' is equal to 75, then enter the number 75 in the Value field.

The image shows a configuration window titled "Range". It contains a radio button that is selected, followed by the word "Range". Below this, there are two input fields. The first is labeled "Min" and contains the number "0". The second is labeled "Max" and contains the number "50".

Use the Range option when you want to match a range of values against the Select action's value. The Min field will contain the minimum value that you want to set for the range and the Max field will contain the maximum value you want to set for the range. If the value of Select action falls within or equals the Min and Max field values then this Case action is activated and the actions listed under this action will be run. To continue with the previous example, you could set the Min value to 0 and the Max value to 50 and if the Select action value for the variable 'myFinalScore' was any number between this range the Case action would be activated.

Finally, in the example given, neither of the Case actions match the value of the Select action because in the first Case the value is too low and in the second Case the value is too high, therefore, the Default action is run.

Information Tab

The Information tab in the **Publication Properties** dialog allows you to store notes about the author and publication as well as automatically tracking how long the publication has been open in the Opus Editor.

The Information tab is useful for describing the contents of the publication or reminders about the process of creating it.

For professional users of Opus building publications for clients, this tab is an invaluable project management tool. The timer allows you to accurately estimate how long you have spent developing a project. This will allow you to provide realistic developments costs as well as more accurately plan timescales for future projects.

Note: The Information tab, combined with the Notes View on each page makes the task of planning and managing your publications much easier – see Notes View in the Help file and *User Guide* for more information.

To edit the Information tab settings:

1. Use the **Author** box to enter the developer of this publication. Click on the **To Notes** button to add the entry in the Author box to the Notes box.

Note: Opus may have already placed a name in the Author box because it checks who is the registered Windows user on your computer and inserts that name into the box. If this isn't your name or you want to enter a different author, a group of authors or a company name, simply delete the current contents and type the new information.

2. The **Editing Time** option shows how long the publication has been opened in the Opus Editor.

The time is cumulative, in other words, every time you re-open the same publication the time is the total amount of time the publication has been opened in the Editor since it was first created.

3. Click the **Reset** button if you want to reset the Editing Time option back to 0 minutes. This is particularly useful when developing large multimedia publications, you can reset the time for each new phase

of the project e.g. after the prototype, design-build, final build and testing phase.

Before you click the Reset button you should add the current Editing Time to the Notes box by clicking on the **To Notes** button. This will add the current time on a new line in the Notes box. Then it is a simple matter of adding all of the times that appear in the Notes box together to get the total time taken on each publication.

4. Use the Notes box to enter any information you want regarding this publication. You can type directly into the box or insert the Editing Time or Author name using the appropriate **To Notes** buttons.

A scrollbar will automatically appear on the right-hand side of the box when you reach the bottom line of the box and press the Enter key to add a new line.

5. Click the **Print Notes** button to print a permanent hard copy of your notes – the standard Windows Print dialog will appear when this button is pressed.

6. Click on the **Apply** button to save your changes.

Permissions

This is a versatile feature of Opus and its benefits are often overlooked. The tab appears on the Properties dialog for any object and allows you to lock the object or specific aspects of it – so that it cannot be changed in the editor. Thus you can design a publication in Opus and pass it on to someone using Opus Presenter (for example) and they can only update the objects you have given them permission to edit.

In addition to the obvious benefits in workgroup computing this has some additional uses...

For businesses this is useful, for example, where a presentation is designed to be customised by sales staff to meet local needs or to be focussed in a particular direction for a particular customer but where key elements need to remain unchanged. This is even more important where compliance with regulations is important because you can ensure each presentation includes the obligatory information.

For educators and trainers it allows you to create resources for students to use within Opus which can only be edited to a certain degree.

For developers it allows you to supply “editable” resources. You can provide your clients with a finished publication and the original Opus file for them edit certain elements, while you retain control of the key elements or the particular design. The ability to offer product which the client can update without recourse to the designer (until it requires more extensive and worthwhile reworking) can be invaluable if handled properly. If your client does not have Opus you can buy special versions at special discounts to distribute the editor with your publications.

Overview

The Permissions tab in the Properties dialog box will appear for every object that you add to a page. Every object in your publication can have permissions set so that you can limit the Interactions and Properties of an object.

- (i) The Interaction permissions – are the ability to Move, Resize, Copy, Delete or Edit an object.
- (ii) The Properties permissions – are the ability to hide the Background, Border, Image, Transitions and Cursors tabs inside the objects Properties dialog.
- (iii) The Lock Out option – this allows you to completely lock an object from being changed in any way.

Note: The permissions only affect objects in the Opus Editor and do NOT affect the way objects work in the published version of the publication.

Password Protection

Although the Permissions settings are not available in non-Professional versions of Opus you may want your colleagues or students to use the publication in Opus and still not be able to change the Permissions settings to allow themselves to edit objects.

By using the password protection option you can set any Permission tab to require a password before editing of the Permissions can be undertaken. Just don't forget your password or you will not be able to change the setting yourself.

Using the Permission Settings

The **Disable Permissions** option allows all editing of an object. When you first create an object, this option is always on (for obvious reasons) and you can change any part of the objects behaviour.

Lock Out

The Lock Out option lets you disable all **Interactions** and **Properties** for the object. This is particularly useful when you have made changes to an object and then want to lock the object so nobody else can change the object. A Permissions Warning box will appear when you attempt to open the Properties dialog for a locked object.

Note: When an object is **locked out**, only the Permissions tab will appear in the Properties dialog for the object in Opus – this allows you to

unlock the object if required. When the publication is opened in non-Professional versions of Opus, a locked object cannot be unlocked.

Custom Settings

Use the **Custom** option if you want to disable a specific combination of **Interactions** and **Properties**. Selecting this option will highlight the options in the Custom panel. Allowing you to tick the objects you want to allow editing of for both Interaction (moving and resizing the object) and/or the Properties of an object (border, effects etc)

Note: A Permissions Warning box will appear when one or more permission has been disabled and you open the object's Properties dialog.

Interactions

Use the **Interaction** to be allowed box to disable or enable the Interactions permissions. There are five types of interactions you can set for every object, an object can be set to: Move, Resize, Copy, Delete or Edit. By default, each of these options is enabled, in other words an object can be moved, resized, copied, etc. To disable an option, remove the tick beside the name of the option you want to disable.

Properties

Use the Properties to be set box to disable or enable the Properties permissions. There are five tabs in an object's Properties dialog that you can hide when the dialog is next displayed, they are: Background, Border, Image, Transitions and Cursors. By default, each of these options is enabled, in other words the tabs appear when an object is first created. To disable an option, remove the tick beside the name of the Property tab you want to hide.

Remember, when you create a new object in Opus, all of the permissions are enabled. You only need to remove permissions when you want to restrict the objects position, size, cut, copy and delete interactions or if you want to remove specific Property tabs from appearing in its Properties dialog.

Index

A

Actions

- Clock · 91
- Conditional Actions · 101
- Database · 29
- File Management · 95
- Launch · 87

Additional Resources · 75

AND · 26

Array

- elements · 61

Arrays

- definition · 61
- populating · 62
- using · 63

Author · 105

C

Case and Select · 103

Clock · 91

- Continue · 93
- Pause · 93
- Start · 91
- Stop · 92

Conditional Actions · 101

Consolidate Resources · 51

Continue Clock · 93

Copy File · 97

Create as Table · 67

Custom Action · 83

custom presentations · 107

Custom Trigger · 83

customisable publications · 107

D

Data Paths · 54

Data Source Names · 20, 21, 22, 29, 30, 35, 79

Creating a New DSN · 20

Re-using DSN's · 21

Database

Actions · 29

Connecting to · 19

Getting Started · 13

Making them Interactive · 17

Overview · 11, 13, 79

Publication Properties · 78

Showing Records · 15

Database Query · 29

Creating a · 23

Delete File · 98

E

editable publications · 107

Editing Time · 105

education · 107

elements · 61

F

File Browse · 95

File Management · 95

Filter Resources · 47

I

Information tab · 105

L

Layout · 71

LIKE

 The LIKE Operator · 25, 27

List Boxes

 selection style · 66

ListBoxes

 selection index · 66

Lock Out · 109

M

Minimise options

 Minimise to System Tray · 82

 Run Minimised · 82

O

ODBC drivers

 Connecting to a Database ·
 19, 20

OR · 26

P

Pattern Matchin · 28

Pause Clock · 93

Permissions · 107

 custom · 109

 Interactions · 109

 password protection · 108

 Properties · 109

 settings · 108

Post Web Data · 87

Preview pane · 44

project estimation · 105

Publication Properties

 Additional Resources · 75

 Database · 78

 Layout · 71

Purge Resources · 56

R

Resource Manager

 overview · 9, 43

 workspace · 43

S

sales presentations · 107

Select and Case · 101

Set\Clear Resources · 49

SQL Wizard · 23

Start Clock · 91

Stop Clock · 92

T

Tables · 67

training · 107

Triggers · 83

U

URL · 87

using Arrays · 63

