TaskCentre® v4.0
Product White Paper

Introduction

Product Overview

Orbis TaskCentre is a unique software suite that enables organisations to build powerful automated business processes whilst leveraging the power of existing systems. The product range offers a totally scalable, cost-effective automation solution for organisations ranging from the SMB to the large Enterprise.

Orbis TaskCentre provides unrivalled functionality for the provision of operational alerting requirements providing scheduled or real-time exception reporting in conjunction with Financial, ERP, MRP, Production, Stock, HR & Project systems to name but a few. In addition, its extensible process modelling capabilities enable the replacement of complex manual tasks, thus driving operational efficiency throughout the enterprise and providing a common automation framework across the entire information chain.

The product is built around the central concepts of Tasks and Steps to provide the ability to build ultimately flexible automation processes with logical building blocks. These processes integrate seamlessly with existing information sources, applications, infrastructure and communications without the need for complex programming.

Tasks

Tasks are the primary entity within Orbis TaskCentre® and represent a part or all of a distinct business process containing multiple interrelated Steps.

By building Tasks it is easy to design business rules that partially or completely replace a manual process. Within each Task there is complete control for which Steps are utilised and in what sequence using visual modelling techniques. A Task can be scheduled to run periodically or triggered real-time via SMTP and MS SQL Server Trigger Steps or through the Task API.

Tools and Steps

Tools are used to create Steps within a Task, providing the functional building blocks that interface with common technologies, systems and applications. They are then joined together in a logical sequence to build the business process. Tools can be grouped into the six categories of Event, Input, Format, Output, Execute and General.

Type	Summary
Event	Triggers a Task to be run and exposes data through variables to other Steps within the Task.
Input	Pulls information into TaskCentre from a source such as a relational database (RDBMS), file format or other structured source and exposes this to other Steps.
Format	Consumes and formats information of one type and then exposes that information as another type to other Steps.
Output	Consumes information from Input or Format Steps and outputs the information into another system, application or communication method.
Execute	Executes an external application, procedure or object, allowing developers to control the precise result of this Step when utilising proprietary systems.
General	Provides specific functionality that cannot be included in the other Tool types or is related to third party applications used with TaskCentre.

Operational Schematic

Figure 1 below illustrates TaskCentre from an operational standpoint. It shows how it provides automation and alerting services and the main components contributing to those services. Furthermore, the diagram demonstrates how TaskCentre allows an organisation to interact with business partners, individuals and systems whether internal or external to provide greater efficiency and cohesion.

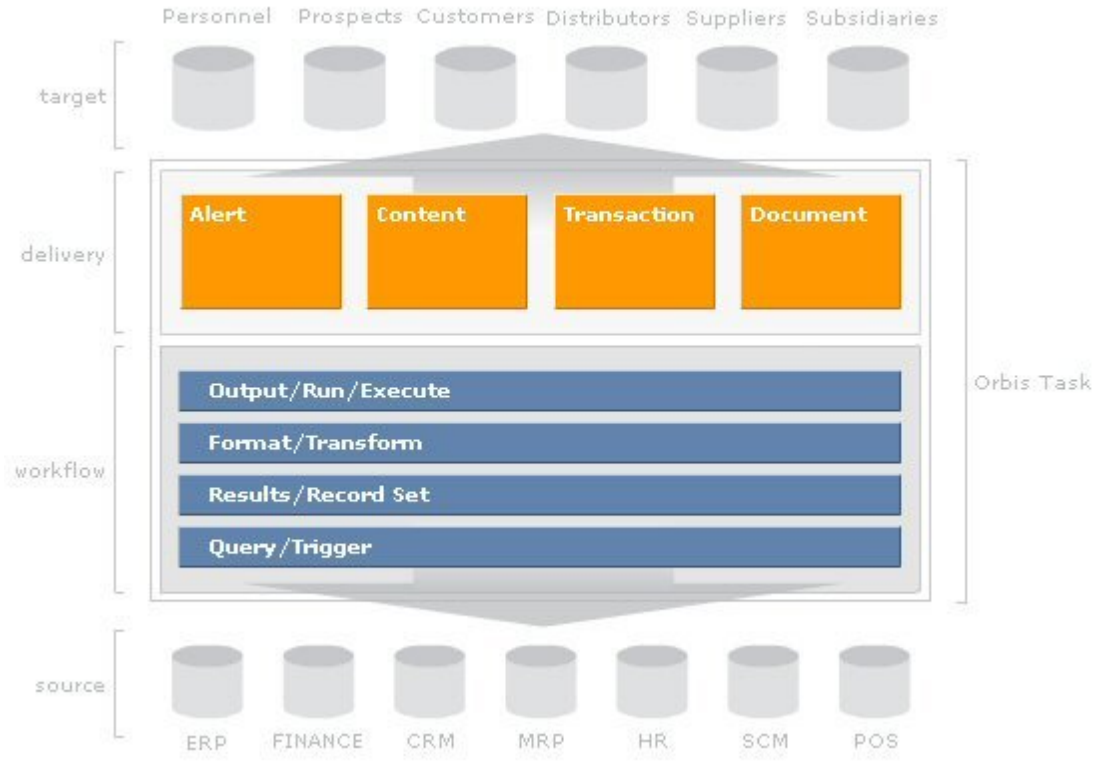


Figure 1 – Operational Schematic

Architecture

Summary

Orbis TaskCentre is 32-bit Windows based software, utilising a true three-tier client server model over TCP/IP, with a multi-threaded server running as an NT Service. The Client provides all administrative and Task design capabilities whilst the Server provides the client connectivity and manages the processing of Tasks.

Resilience

The TaskCentre Server utilises out-of-process COM Server technology to isolate a Task instance from both the TaskCentre Server itself and other Task instances. This means that external inconsistencies such as exceptions caused by ODBC drivers, applications or other system APIs do not affect ongoing TaskCentre operations beyond that Task instance. Furthermore, such untoward occurrences are logged immediately by TaskCentre and notifications are sent to the administrator and Task 'Owner'.

Asynchronous Processing

The multi-threaded TaskCentre Server enables multiple Tasks to run simultaneously to avoid queuing. The number of Task threads can be controlled by the administrator depending on the processing load.

Performance

TaskCentre demonstrates ultimate performance due to its highly-optimised, small-footprint architecture and use of cutting-edge development techniques which optimise multi-processor support and minimise processor context-switching.

Tasks

Summary

Fundamentally, the main Task interface provides a visual canvass on which the user defines the process using the Tools provided to create Steps. The sequence of Steps and flow of data within the Task is completely within the control of the user.

Design

Steps created using the Tools can be dragged and dropped onto the Task Plan canvass from the Task Browser dialog and then arranged and sequenced, again using drag and drop. Each Tool interface allows precise configuration of a Steps behaviour within a Task (See 'Tools' below for more detail on specific Tools).

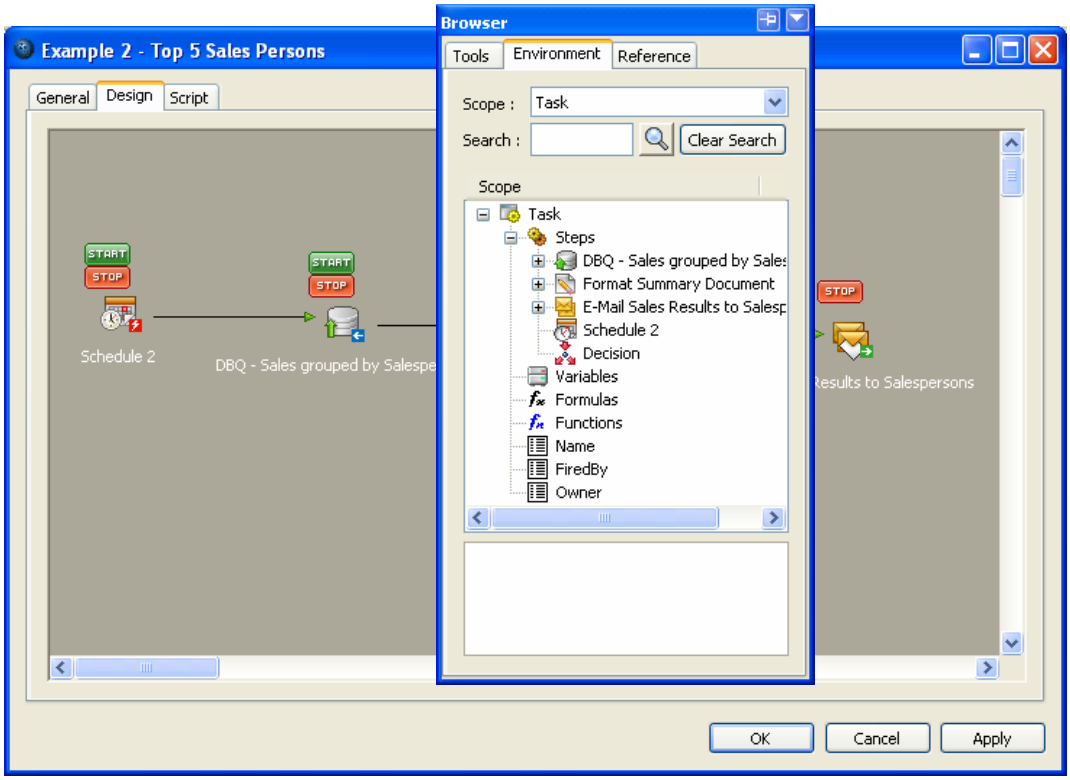


Figure 2 – Task dialog – Design tab and Task Browser

Script

Through the Script tab the actual SQL script created for a Task through selections made in the Design tab is displayed. If preferred, a Task designer may chose to create the script manually or edit the existing script.

A useful feature of creating a Task in script mode is that any variables, functions, formulas and Step properties that have been created may be dragged and dropped directly into the script from the Task Browser dialog.

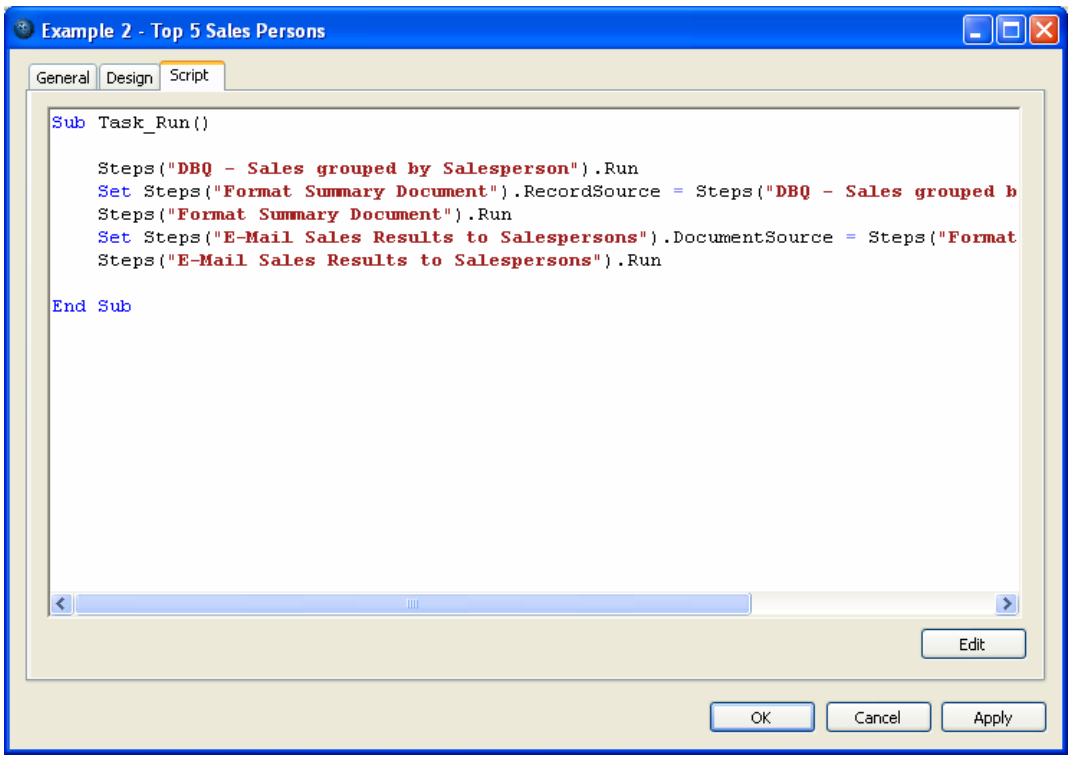


Figure 3 – Task dialog – Script tab

Global Options

Server

This feature allows a system administrator to configure how the TaskCentre server 'listens' for connections from TaskCentre clients and how many Tasks may be run asynchronously on the server.

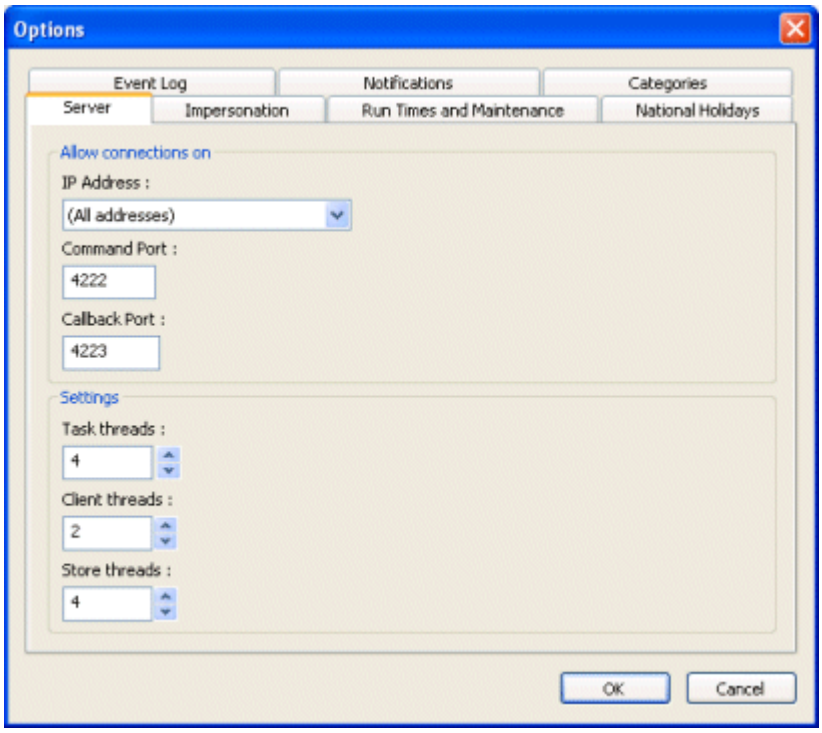


Figure 4 – Options dialog – Server tab

Impersonation

The impersonation feature may be used to run all Tasks under a specific domain users credentials to enable them to 'Impersonate' that user and therefore gain access to a database via a 'trusted' connection by specifying their credentials.

Specifying a user account in this tab will override the account used on the TaskCentre service which by default is the local system account.

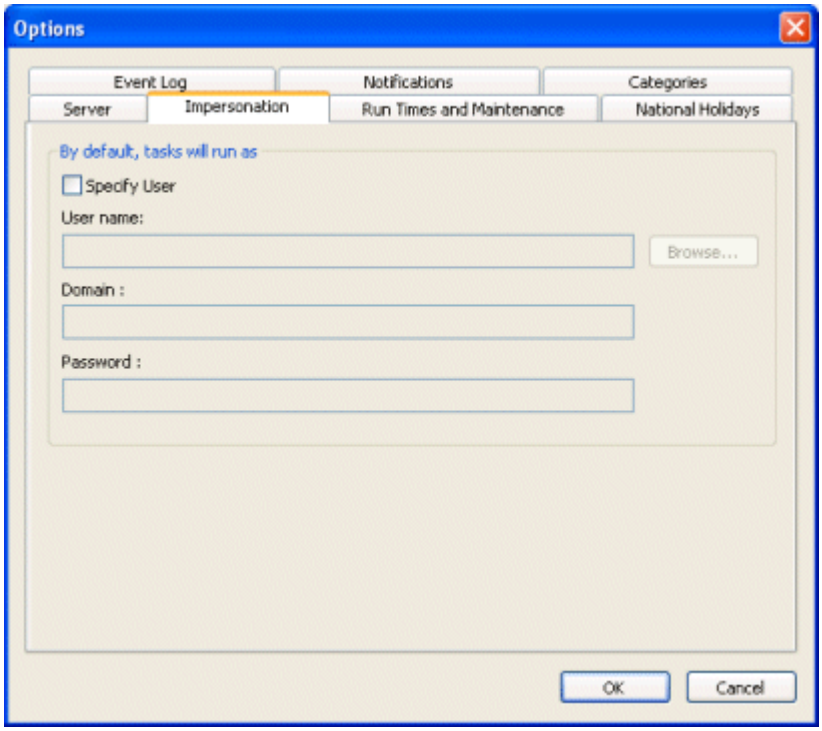


Figure 5 – Options dialog – Impersonation tab

Run Times and Maintenance

The Run Times and Maintenance tab of the Options dialog is used to globally set times of the day and days of the week that Tasks may be run and times when maintenance routines will automatically be run. This enables the user to ensure that Tasks do not conflict with other automated processes such as backup or data warehousing routines.

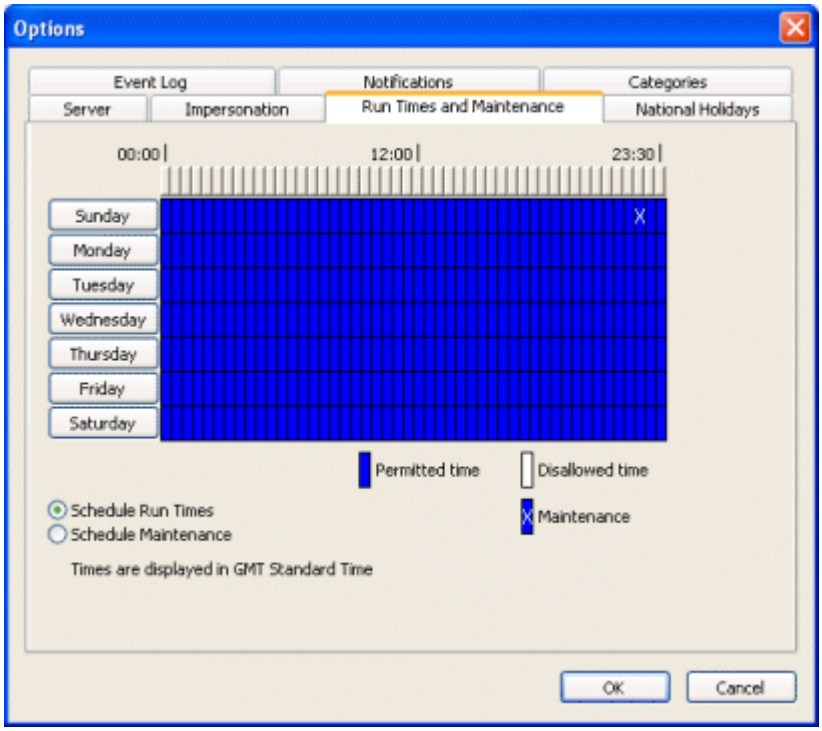


Figure 6 – Options dialog – Run Times and Maintenance tab

National Holidays

This feature is used to set specific dates when Tasks are not to be run. To enable a list of dates to be quickly compiled, an import feature is provided through which a pre-configured list of dates for a specific country may be automatically added. Alternatively, dates may be individually added and then edited or removed as required.

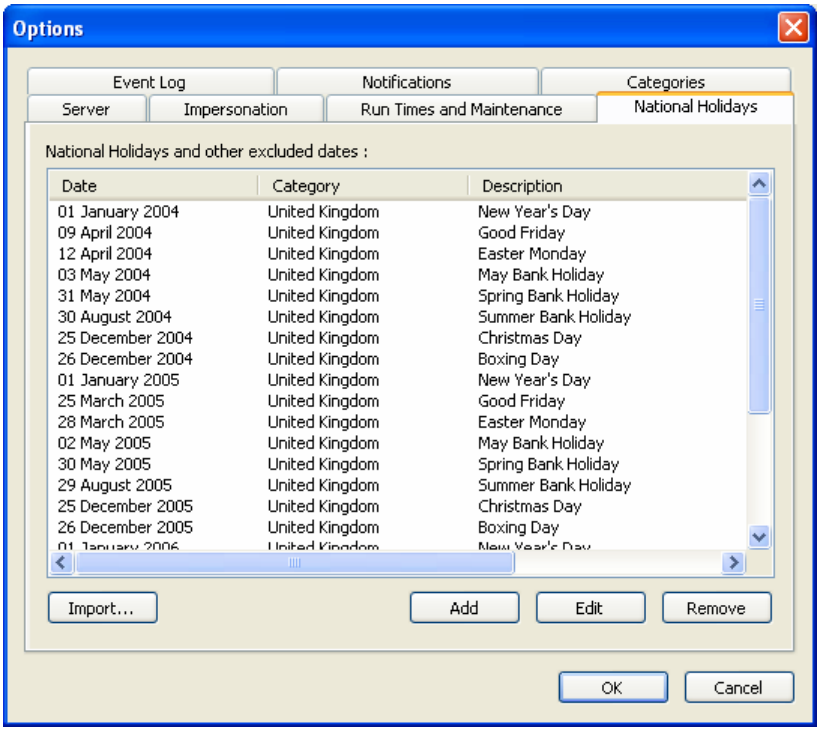


Figure 7 – Options dialog – National Holidays tab

Event Log

This feature provides a system administrator with the facility to set a global time interval in days after which event log entries are purged.

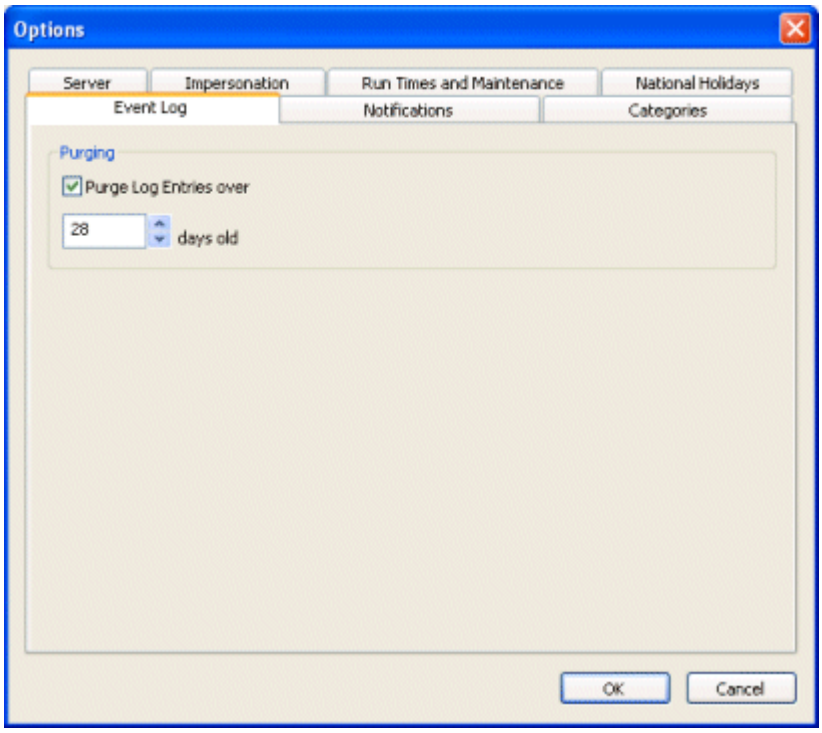


Figure 8 – Options dialog – Event Log tab

Notifications

This feature provides the facility to set up automatic notifications by e-mail to System Administrators and Task owners when selected warnings and errors occur within TaskCentre.

An administrator can configure exactly what notifications are required for the different types of potential occurrences. Administrators and Task owners know the instant that something has gone wrong so that they can assess the situation and take remedial action if required.

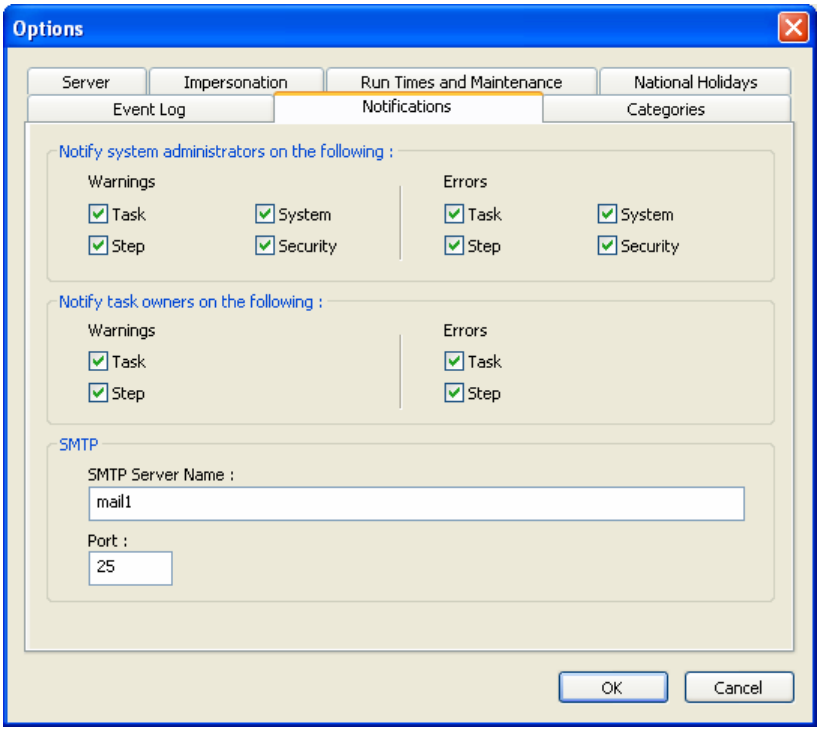


Figure 9 – Options dialog – Notifications tab

Categories

This feature provides the facility to create Category names which may then be assigned to Folders or Tasks to group common items together.

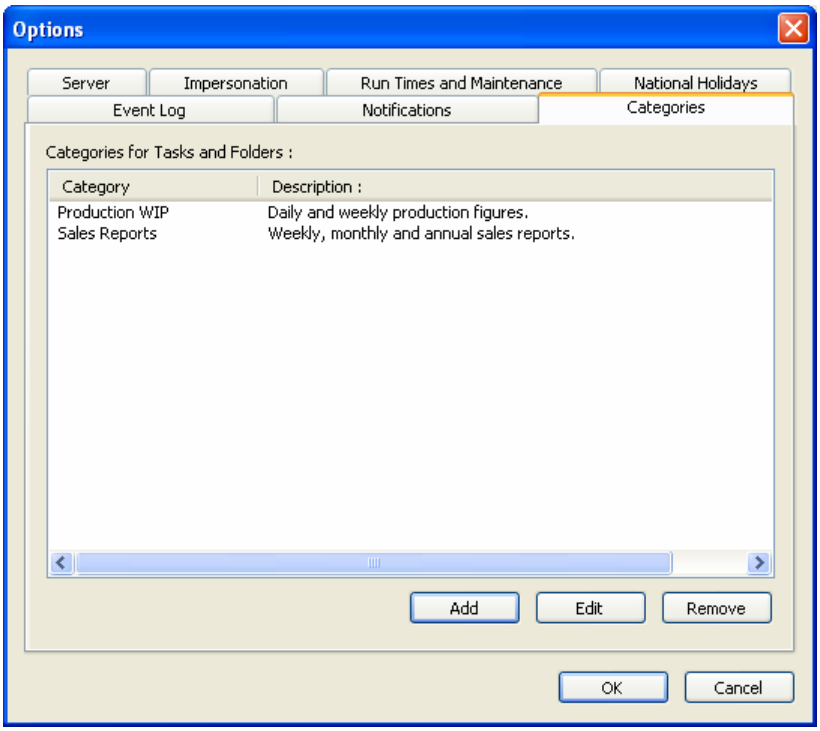


Figure 10 – Options dialog – Categories tab

Task Options

Run Times

In addition to the global run times and maintenance options, further run time restrictions may be defined for each Task as shown in the example below. In the example the Task will not run at all during the weekend and between 9.30PM and 12.00AM during weekdays. The X in the block for the last half hour of Sunday indicates when the automated system maintenance routine will run.

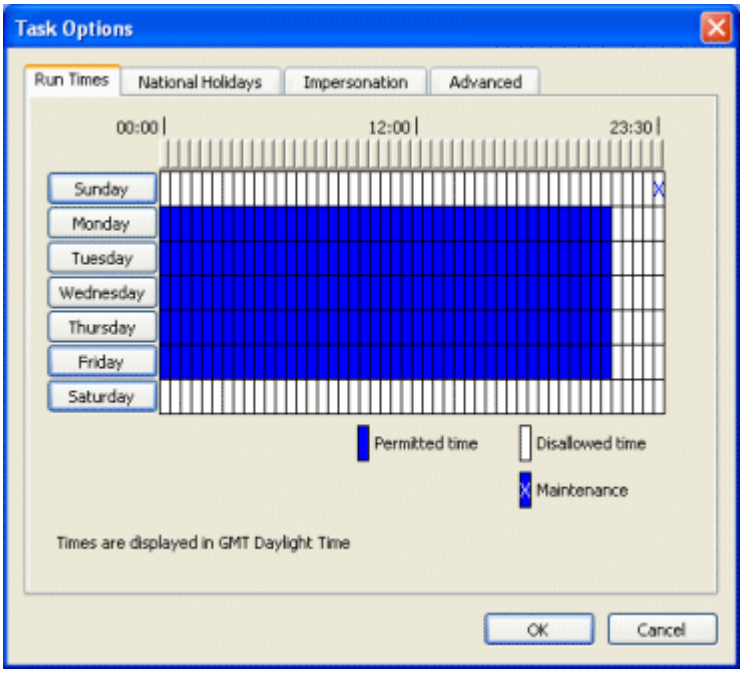


Figure 11– Task Options – Run Times tab

National Holidays

To allow complete flexibility on Task run times, this feature provides each Task with the option to over-ride any global settings by excluding any or all specified dates for that Task and so allow it to run.

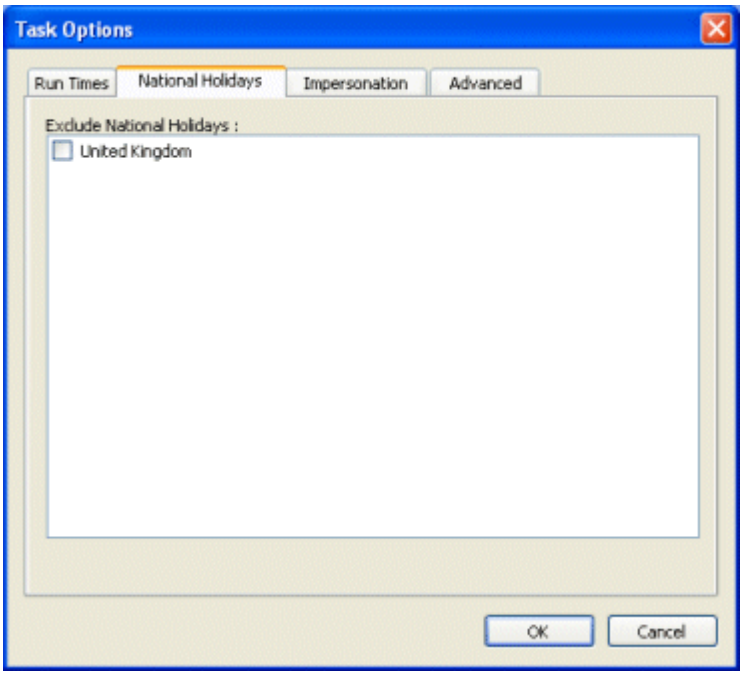


Figure 12 – Task Options – National Holidays tab

Impersonation

Global setting are used to configure all Tasks to be run using a specific domain users credentials. To allow complete flexibility, this feature provides each Task with the option to specify any available domain user to enable the Task to 'Impersonate' that user.

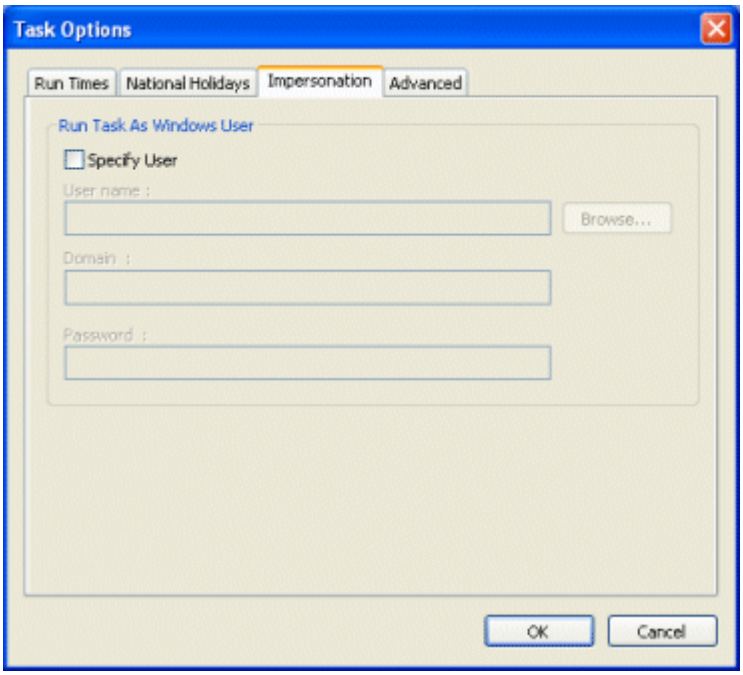


Figure 13 – Task Options – Impersonation tab

Advanced

The Advanced tab displays the Tasks' unique ID, which is particularly useful when using the Task API, and is also used for setting the maximum permissible number of concurrent instances of the Task.

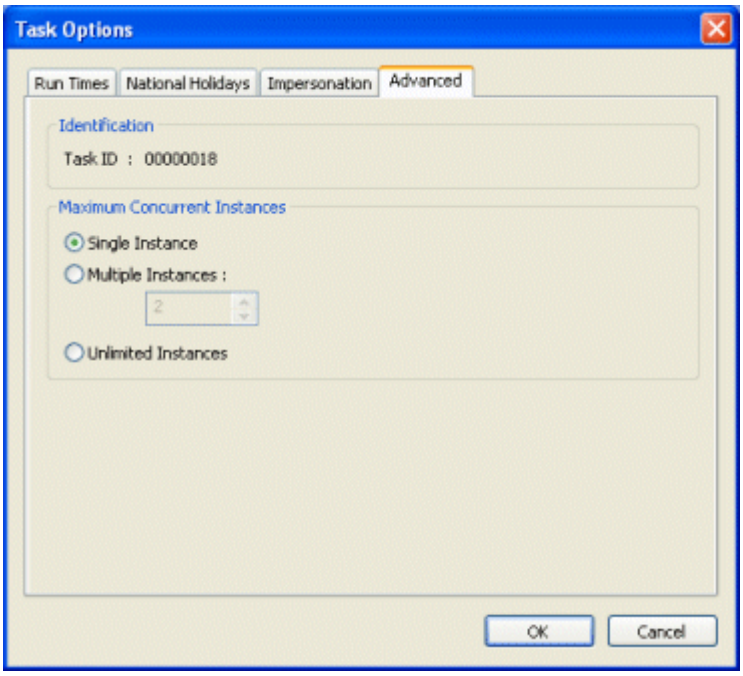


Figure 14 – Task Options – Advanced tab

Task Browser

The Task Browser dialog is displayed whenever the Task dialog is open and is a powerful tool for managing and utilising all data and information related to a Task through a number of tabbed panes.

Tools

The Tools tab exposes all available Tools which may be selected and used to create and add Steps to a Task.

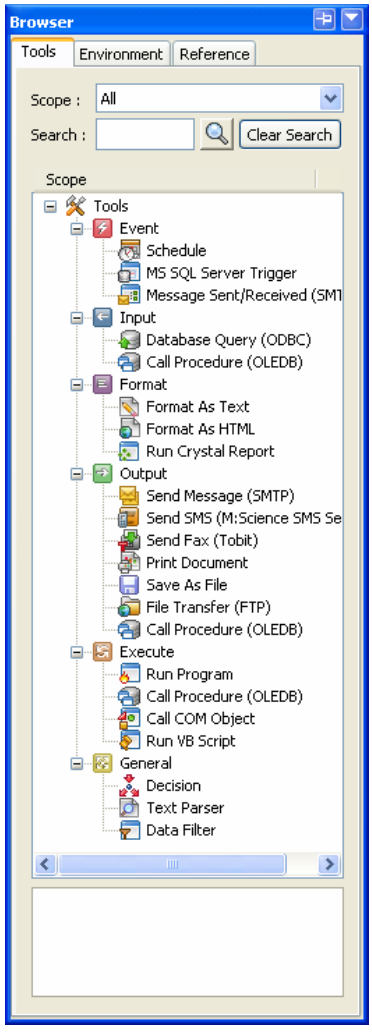


Figure 15 – Task Browser – Tools tab

Environment

The Environment tab exposes the data for all Steps created within a Task and for the Task itself. In addition, variables, formulas and functions may be created for use within Task Steps and the Task script.

Also, some Steps provide the facility to map data to variables which may then be utilised to display the data through other Steps, either programmatically or by dragging and dropping the variable into an html or text page.

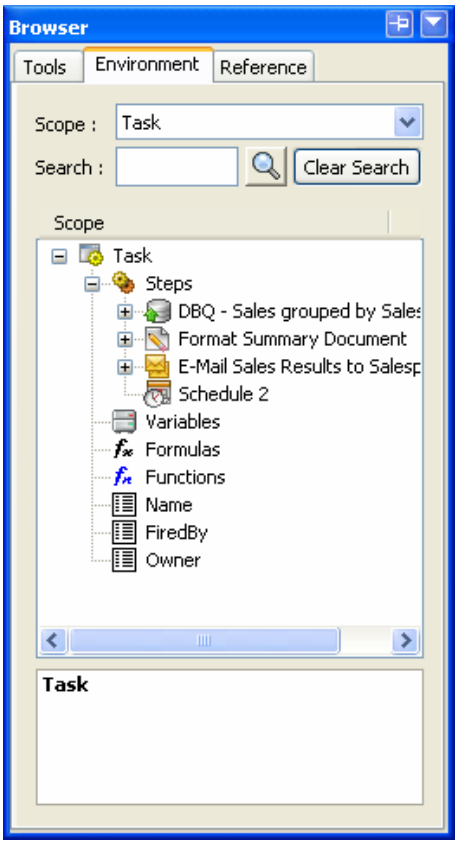


Figure 16 – Task Browser – Environment tab

Reference

The Reference tab exposes a number of common scripts which may be utilised within Task Steps by dragging and dropping the script into a field within a configuration dialog or into the Task script itself when creating the Task script manually.

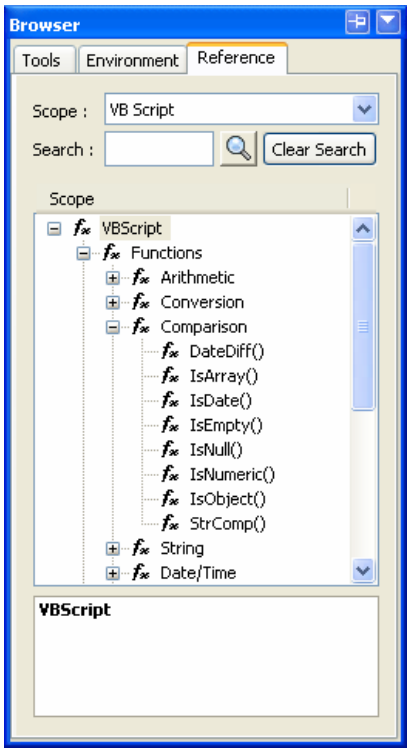


Figure 17 – Task Browser – Reference tab

Tools

Summary

As described earlier, Tools are used to create Steps within a Task which are then linked together to form the 'Task Plan'. Steps expose and consume information to and from each other in different ways depending on their specific purpose.

Within every Tool there are individual properties and settings to accomplish what is required from the Step that it creates and therefore the Task as a whole. By using combinations of Steps, an amazing flexibility of automation is possible including Alerting, Request/Response, Web Publishing, Automated Document Distribution and fulfilment e.g. Order Acknowledgements, Statements to Fax or Print, System Monitoring, Replication, Announcements, Data Extraction, Integration etc.

As previously explained, in general Tools can be grouped into Event, Input, Format, Output, Execute and General categories as shown in the table below.

Event	
	Schedule
	MS SQL Server Trigger
	Message Sent/Received (SMTP)

Input	
	Database Query (ODBC)
	Call Procedure (OLEDB)

Format	
	Format As Text
	Format As HTML
	Run Crystal Report

Output	
	Send Message (SMTP)
	Send SMS (m:science SMS Server)
	Send Fax (Tobit)
	Save As File
	Print Document
	File Transfer (FTP)
	Call Procedure (OLEDB)

Execute	
	Run Program
	Call COM Object
	Run VB Script
	Call Procedure (OLEDB)

General	
	Decision
	Text Parser
	Data Filter

Event Tools

Schedule

Summary

The Schedule Tool is used to create a Task Step that triggers a Task to be run, based on specific times and dates. Any number of Schedule Steps may be created for a Task.

However, there are restrictions when Tasks may be run, such as during system maintenance, holidays and other restricted times. These are User configured for a specific Task through the Task Options dialog and for all Tasks through the global Options dialog.

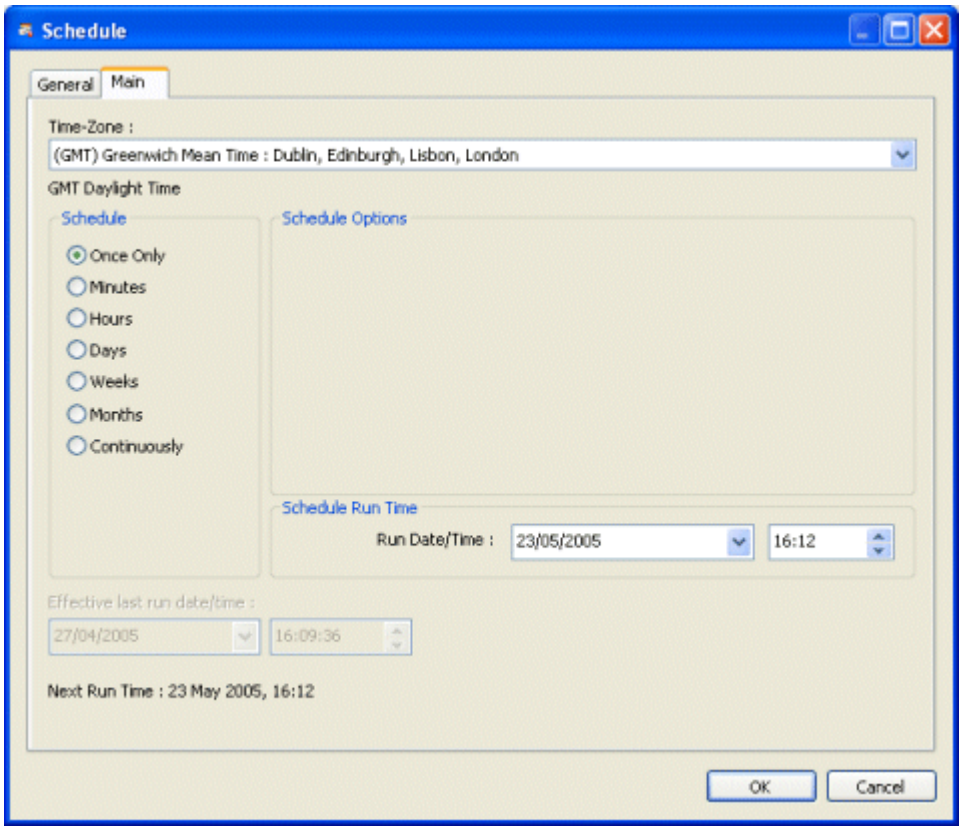


Figure 18 – Schedule – Main tab

Features

- Multiple Schedules may be created for a Task.
- Multiple Time-Zone options are provided to enable the run time of a Task to be synchronised with international operations.

MS SQL Server Trigger

Summary

The MS SQL Server Trigger Tool is used to create a Task Step that utilises the Microsoft SQL Server trigger functionality to create a trigger on a table of a specified SQL data source. When a specified data modification is attempted, such as an attempt to add or delete a table row, the trigger then causes a Task to run.

The trigger creates virtual tables for inserted/updated and deleted rows which may be mapped to Task variables so that the data may be used in other Steps within the Task.

If a Database Query (ODBC) Step is used within the Task to retrieve a recordset from the SQL data source, then the memory feature may be used in a subsequent Step to memorise all of the data into a Repository.

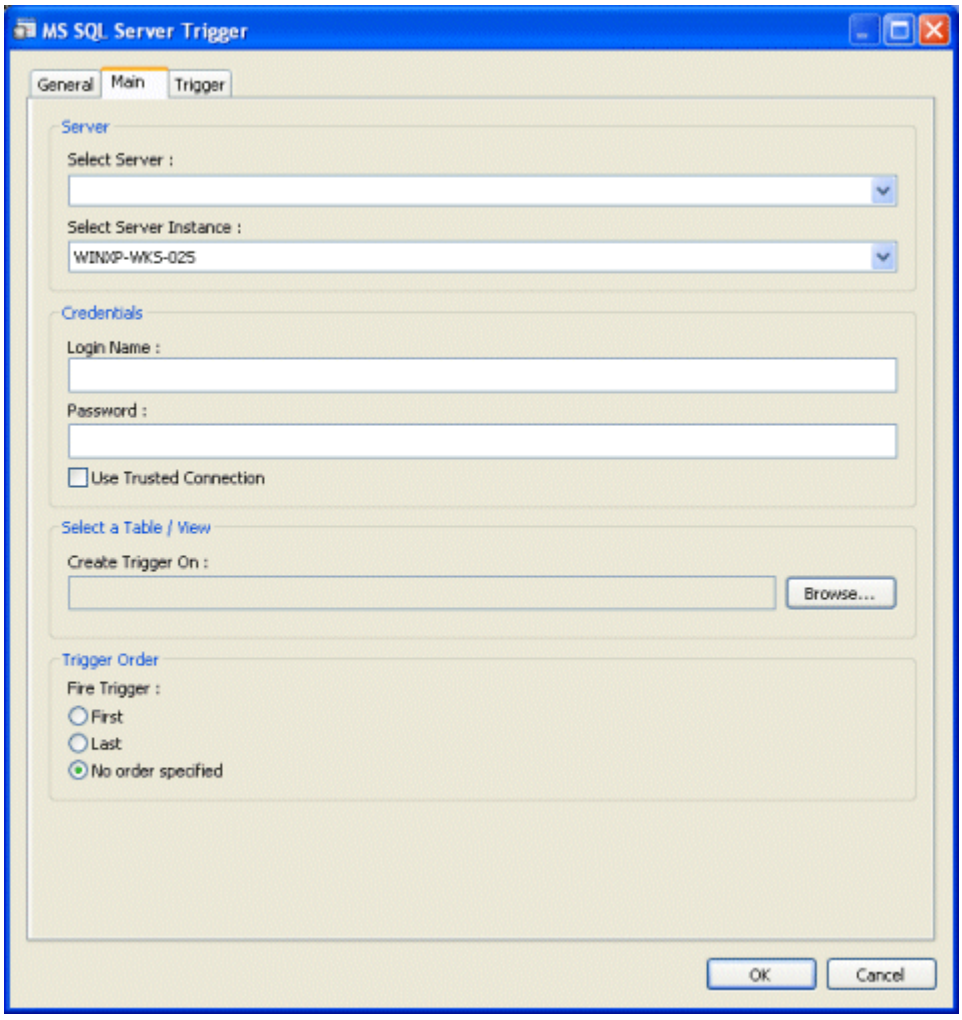


Figure 19 – MS SQL Server Trigger – Main tab

Features

- A Trigger may be activated when any row is updated, deleted or a new row inserted.
- A Trigger may be activated when specified columns are updated.
- Trigger text may be encrypted.
- Values of updated, deleted or new rows in the virtual tables may be mapped to variables for use in other Steps within the Task.

Message Sent/Received (SMTP)

Summary

The Message Sent/Received (SMTP) Tool is used to create a Task Step that triggers a Task to run based on the contents of incoming or outgoing mail sent via a virtual SMTP Mail Server.

When an email is received it is compared against a number of filters created in the Step which define values for the message parameters. Where all filter requirements are met the message is utilised to trigger a Task to run. On receipt, each of the email parameters is exposed and their values may be mapped to variables created for the Task and then used in other Steps.

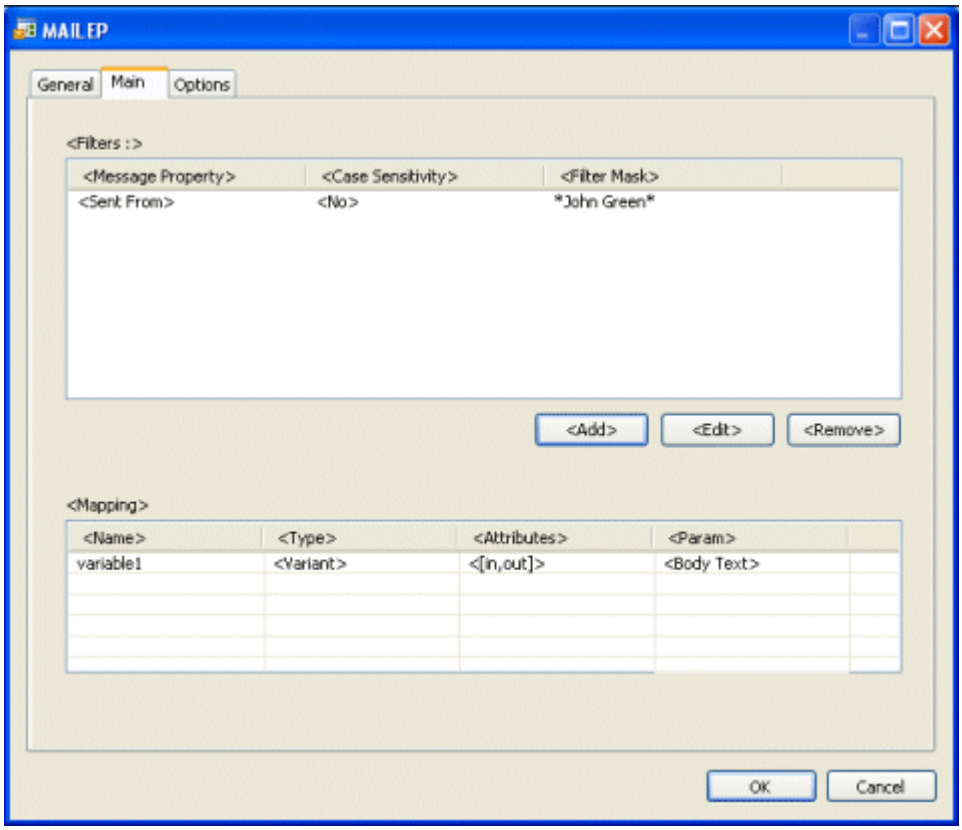


Figure 20 – Message Sent/Received (SMTP) – Main tab

Features

- Create filters for any message parameter
- Include multiple filters in a Step
- Map message parameters to Task variables for use in other Steps

Input Tools

Database Query (ODBC)

Summary

The Database Query (ODBC) Tool is used to create a Task Step that allows the extraction of specific information sets from relational database systems such as ERP, Financial, Accounting, MRP, Production, CRM, HR, Project and Bespoke systems to name but a few. Once obtained, the information is then available to other Format or Output Steps for manipulation and/or delivery. The Database Query (ODBC) Tool has comprehensive ODBC support for both modern and legacy platforms supporting both the SQL 89 and SQL 92 ODBC standards and variations thereof.

The Tool provides a highly functional visual interface for building queries using drag and drop as well as the ability to write completely custom SQL statements if that finite level of control is required. When using the visual method, the SQL is created for you and can be viewed for clarity. Queries can be tested and viewed with small sampling functionality available for efficient testing of larger queries.

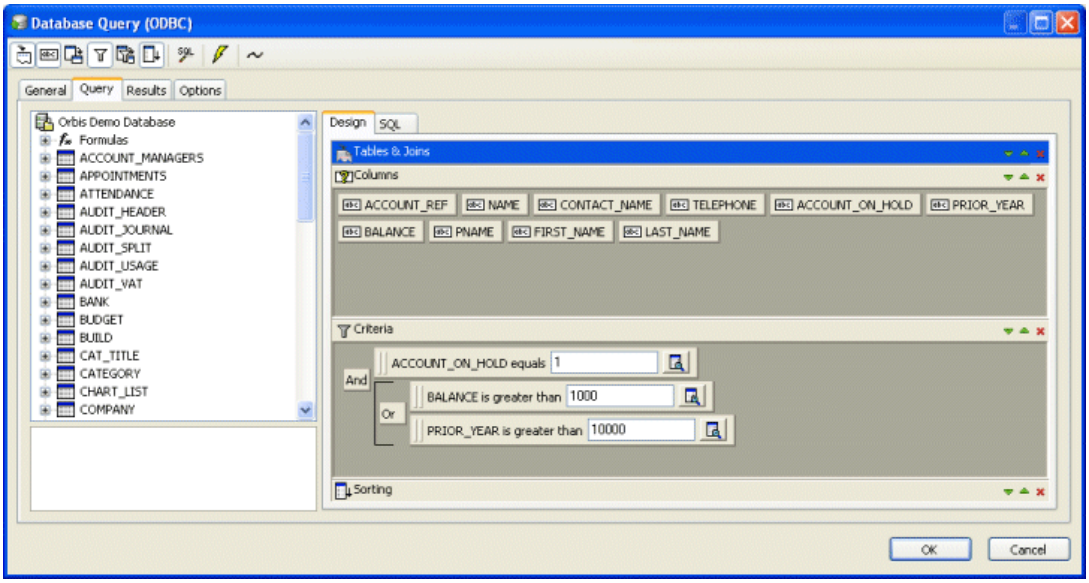


Figure 21– Database Query (ODBC) – Query tab

Features

- Support for SQL 89 & 92 and variations thereof
- ODBC Direct or ODBC Via DAO
- System, User and File DSNs
- Data Models
- Forward-Only or Scrollable Cursors
- Catalogues and Schemas
- Support for all primary SQL clauses

- System Tables, User Tables, Views
- Inner joins, Outer joins, Formula Joins, Cartesian product
- Formula Columns
- Table Aliasing
- Flexible Pre-built Date Criteria
- Drag & Drop Visual Criteria building with nested parentheses (brackets)
- Common and custom Predicates
- SQL Free-type mode

Using Data Models

Many databases use developer notation and naming conventions that to the operational user bear little resemblance to the application itself. In addition, the sheer volume of tables and fields present in most databases can be daunting to even the 'power-user'.

In consideration of these issues, a Data Model provides a semantic layer between the user and the raw data source. This means that if a user is familiar with a given database in an operational sense rather than from a database or development perspective they will still be able to find and manipulate the data they require. A Data Model makes use of user oriented hierarchies containing Groups and Objects, again with user oriented terminology. The following figures demonstrate the difference between using the raw data source and a Data Model for the same data source.

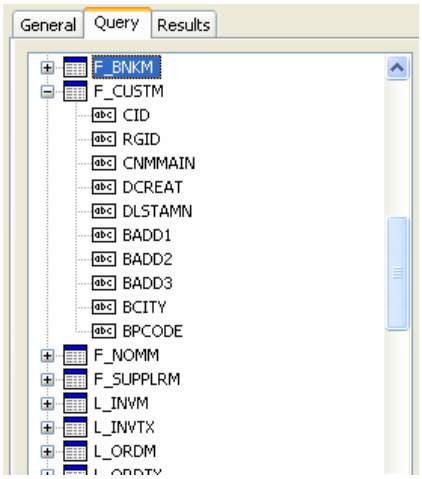


Fig. 22 – No Model

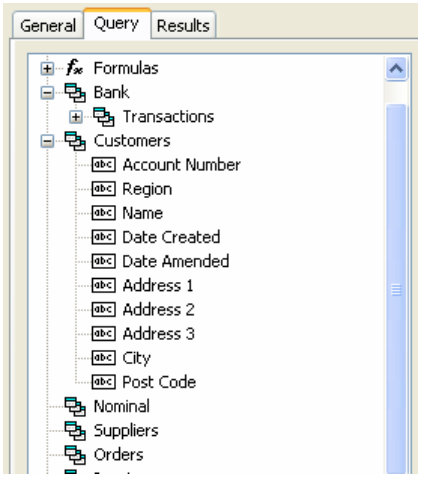


Fig. 23 - Model

Call Procedure (OLEDB)

Summary

The Call Procedure (OLEDB) Tool is used to create a Task Step that allows the execution of database stored procedures and functions to provide read, write and automated data processing.

If a procedure generates a recordset, the Step can return this to other Steps for format and/or delivery. Conversely, the Step can consume a recordset to enable multiple procedure calls or indeed simply execute a single procedure. Procedure calls can be passed static or dynamic parameters in order to control the precise nature of the call being made.

The features provided by the Call Procedure Tool therefore mean that it may be used as an Input, Output or Execute type Step within a Task.

The Tool supports any RDBMS with a functional OLEDB Provider, thus enabling native support for a multitude of data sources.

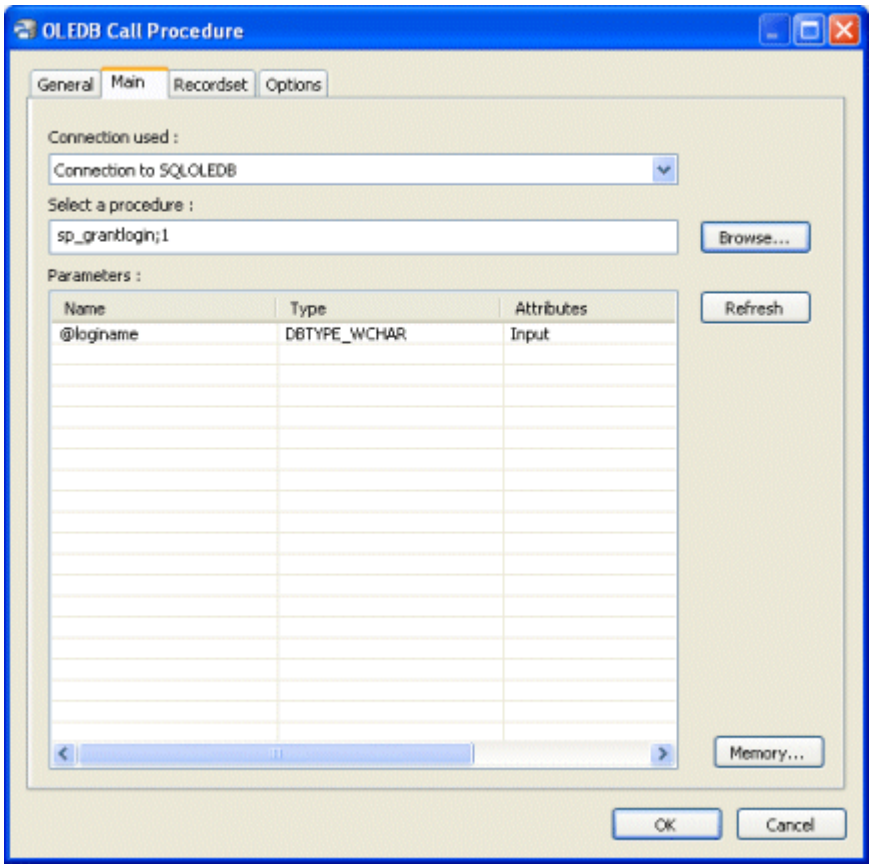


Figure 24– Call Procedure (OLEDB) – Main tab

Features

- Calls stored procedures from a multitude of data sources
- Allows manual or dynamic inputting of procedure parameters
- Contains memory features to allow data to be saved to a Repository.

Format Tools

Format as Text

Summary

The Format As Text Tool is used to create a Task Step that creates single or multiple textual documents, primarily utilising information provided by Input Tools such as the Database Query (ODBC) Tool and requires no global settings.

The Step uses merge fields to merge data objects into the document from a source to produce dynamic documents such as alerts and notifications or data formats such as .CSV for spreadsheets or uploading to other systems.

These textual documents can then be delivered using many of the Output Steps such as Send Message (SMTP) as body and/or attachment, Send SMS (m:science SMS Server), Send Fax(Tobit), File Transfer (FTP) and Save to File.

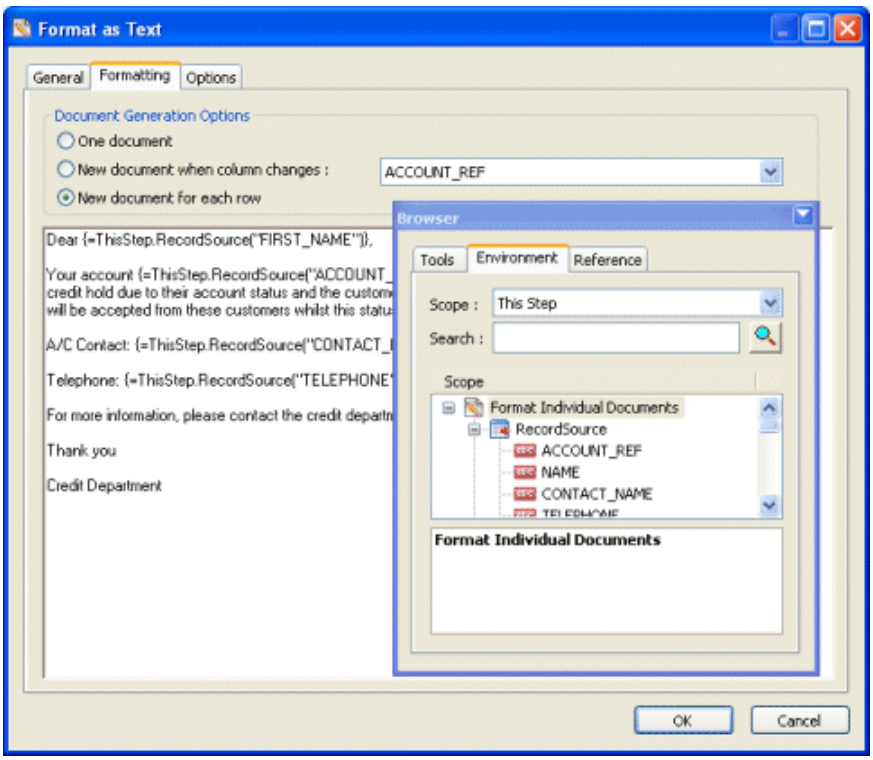


Figure 25 – Format As Text – Formatting tab, Task Browser dialog.

Features

- Single or multiple documents
- New document when data changes or for each row
- Header/Detail/Footer sections
- Merge dynamic data into document
- Produce CSV or other text data formats

Format as HTML

Summary

The Format As HTML Tool is used to create a Task Step that produces single or multiple HTML documents, primarily utilising information provided by Input Steps such as the Database Query (ODBC). You may create new HTML pages within the interface or import pages that have already been created elsewhere. The Step uses merge fields to merge data objects into the document and/or insert tabulated data from a source to produce dynamic documents such as order acknowledgements, statements, intranet or web site content or summary reports. These HTML documents can then be delivered using many of the Output Steps such as Send Message (SMTP), File Transfer (FTP) and Save to File.

The HTML Table functionality is extremely flexible allowing presentation of the data in exactly the form required. Tables support grouping, sub totals, grand totals, data formatting and ultimately flexible hyperlink support for common hyperlink types to enable drill-down/through to other content or custom protocols to automate applications.

Use your default third-party HTML editor such as Microsoft FrontPage in conjunction with TaskCentre by simply using the 'Edit' button.

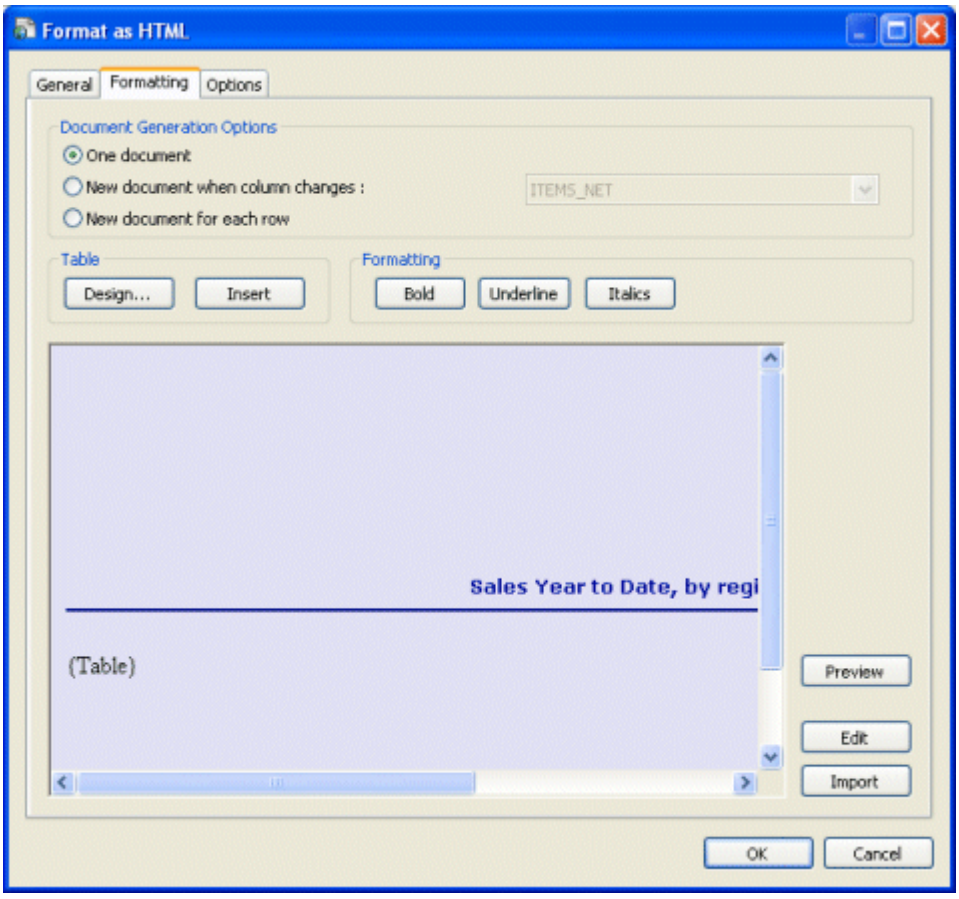


Figure 26 – Format As HTML – Formatting tab

Features

- Create or utilise existing HTML documents
- Merge data from other Actions
- Full support for embedded graphics and backgrounds
- Create single or multiple documents depending on the information
- Default HTML Editor Support from Interface
- Document Preview
- HTML Table Designer
    - Drag-drop table builder
    - Flexible Data formatting
    - Flexible Hyperlink Support
    - Auto Hyperlink detection for dynamic data
    - Flexible formatting of fonts, borders, colours, backgrounds
    - Sub totals & Grand total
    - Flexible total labelling & positioning

Run Crystal Report

Summary

The Run Crystal Report Tool is used to create a Task Step to automate the running of Crystal reports and has no global settings.

By the use of dynamic data passed to the report as run time parameters the Step can be used to run a report one or many times. The dynamic data could be from an Input Step such as Database Query (ODBC).

In use, an example would be running a statement for each of a list of customers, or producing an order acknowledgement for each of the current day's orders.

In addition, where a report requires access to data tables that have specific security associated with them, the relevant login names and encrypted passwords can be passed by the Step to the report concerned.

The documents exposed by the Step can be 'delivered' by Output Steps such as Send Message (SMTP), File Transfer (FTP) or Save as File and then used to present sophisticated management information, delivered via email, fax or published to form part of web or intranet content.

New reports are designed in the Seagate Crystal Reports™ Designer environment, so experienced users of this powerful application will quickly be able to produce a new report, or modify an existing one, and integrate it with TaskCentre.

The inclusion of sub reports is also managed through the provision of a separate report tab for each sub report which provides the same configuration options covering the mapping of parameters to dynamic data from an Input Step and access to secure data tables.

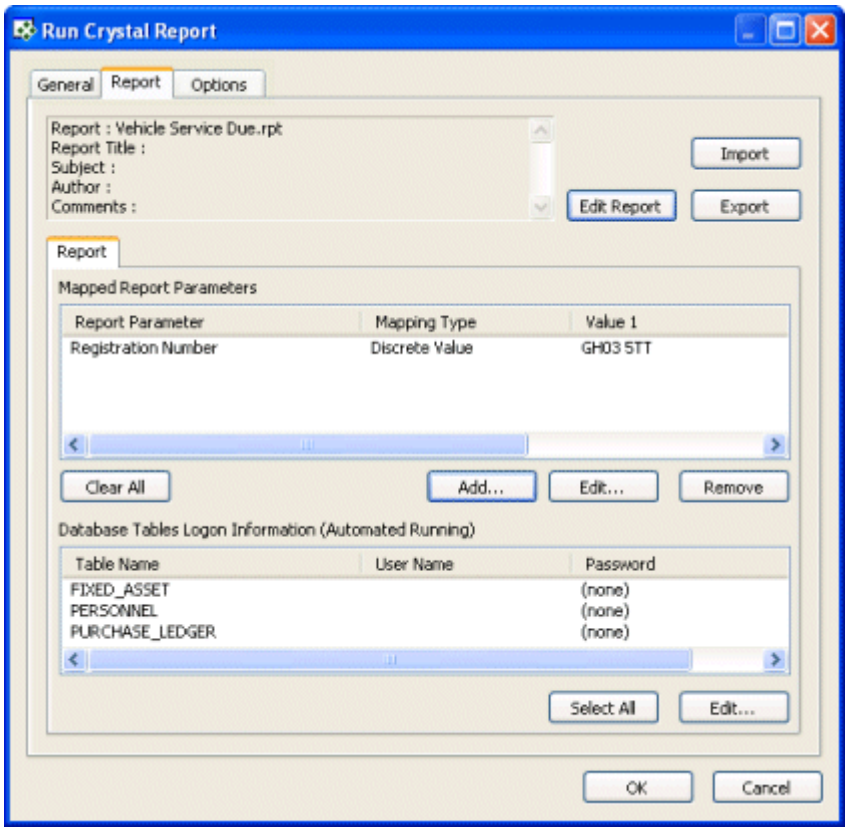


Figure 27 – Run Crystal Report – Report tab

Features

- Crystal Reports accessed via interface within TaskCentre
- All the power of the Crystal Reporting engine at your disposal
- Import and utilise existing Crystal reports
- Map dynamic data to the report and any nested sub reports as run time parameters
- Report Preview in Crystal Reports
- Data security parameters passed to the report at run time

Output Tools

Send Message (SMTP)

Summary

The Send Message (SMTP) Tool is used to create a Task Step that sends messages to any SMTP compliant mail server. The Step is capable of sending multiple messages in either text or HTML formats, incorporating data from an Input and/or Format Step to any number of recipients. If data being used from an Input Step includes a column containing e-mail addresses, then this may be used as a 'dynamic' recipient address.

The Tool is configured globally with the name or IP address and port number of the SMTP Server to use, and this is then transparent for all Steps created. An internal address book is provided so that common SMTP addresses can be defined and re-used.

MAPI address books such as those available from Microsoft Mail, Microsoft Outlook and the Microsoft Exchange Server Global Address List (GAL) are also supported.

Commonly the message content would be a document generated by a Format type Step such as Format As Text, Format As HTML or Run Crystal Report but it is also possible to attach the document to the message and additionally create fixed or dynamic message content by inserting columns from the recordset being used.

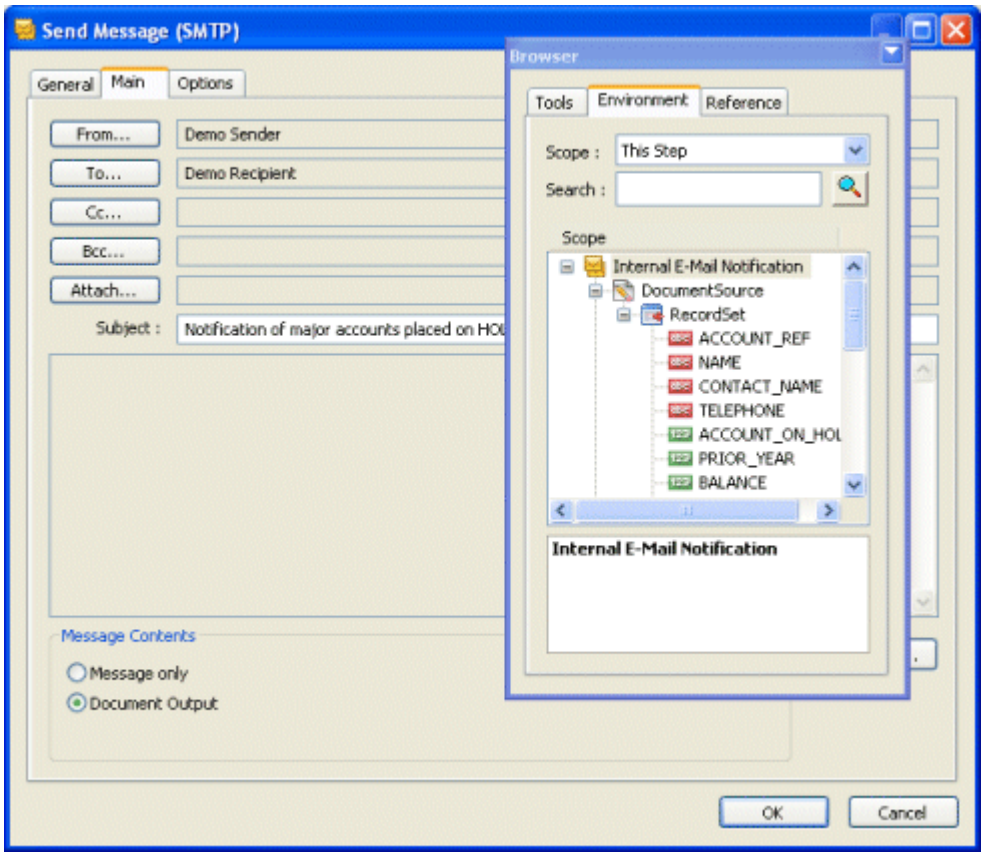


Figure 28 – Send Message (SMTP) – Main tab, Task Browser dialog.

Features

- Native ESMTP client
- Configurable port selection
- Supports From, To, CC, BCC
- Static and dynamic from/recipients
- Supports HTML and text formats
- Memory features to moderate future behaviour based on previous behaviour
- Internal address book
- Support for MAPI address books (requires MAPI)
- Supports static and dynamic attachments

Memory

A Memory feature appears in many Tools, particularly of the Output type, to enable data for selected columns of the recordset produced when a Task is run to be saved into a memory store within TaskCentre called a Repository. See 'Repositories' for further details.

Multiple options are provided to enable a user to specify exactly what is to be memorised and the Repository name and location.

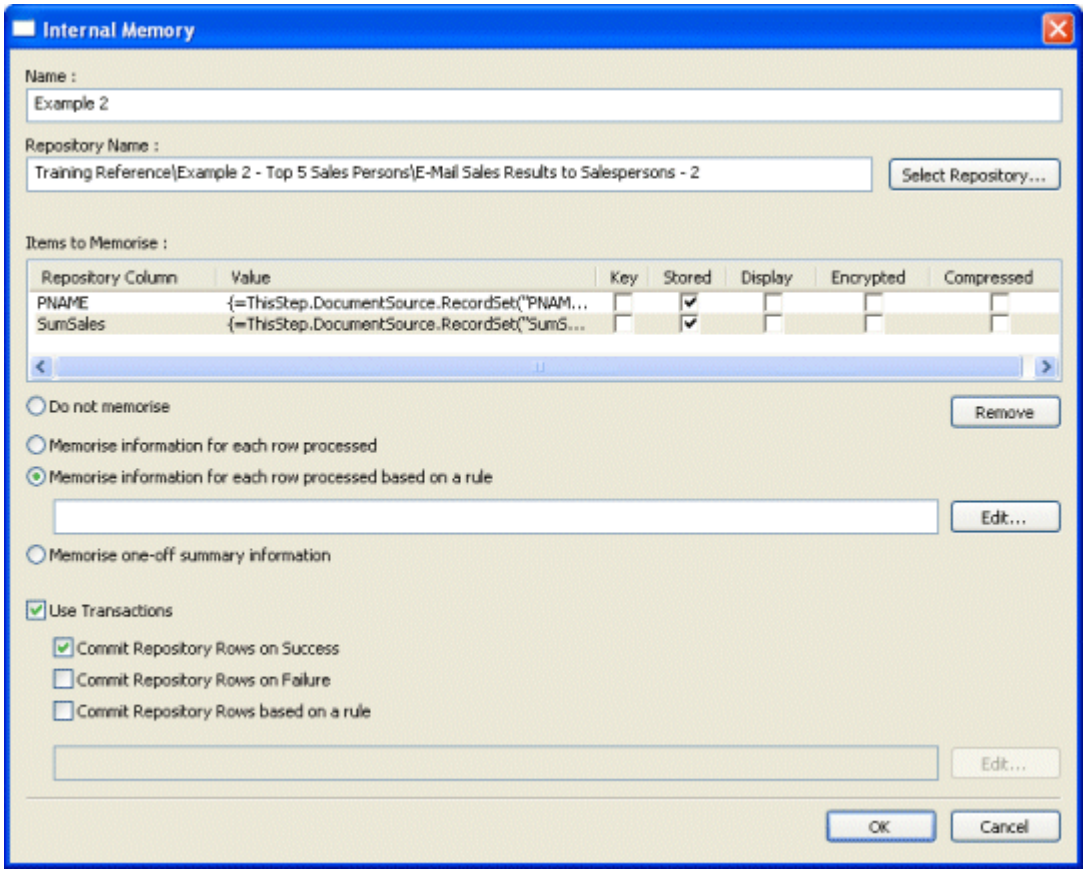


Figure 29 – Internal Memory dialog

Send SMS (M:Science SMS Server)

Summary

The Send SMS (m:science SMS Server) Tool is used to create a Task Step to send SMS (Short Message Service) text messages to one or more GSM mobile devices via the market-leading SMS Server product from m:science.

This product can be reviewed and evaluated at www.m-science.com. Please send an Email or phone 01202 241120 for further information.

The Step is capable of sending static or dynamic SMS Messages to one or multiple recipients. Typically the Step works by using text documents produced by the Format as Text Step due to the nature of the SMS medium. If the source data for the Text documents includes a mobile number, then this may be used as a dynamic recipient address for the SMS messages. TaskCentre simply connects to the m:science product via its built in API by specifying the host server and authentication details.

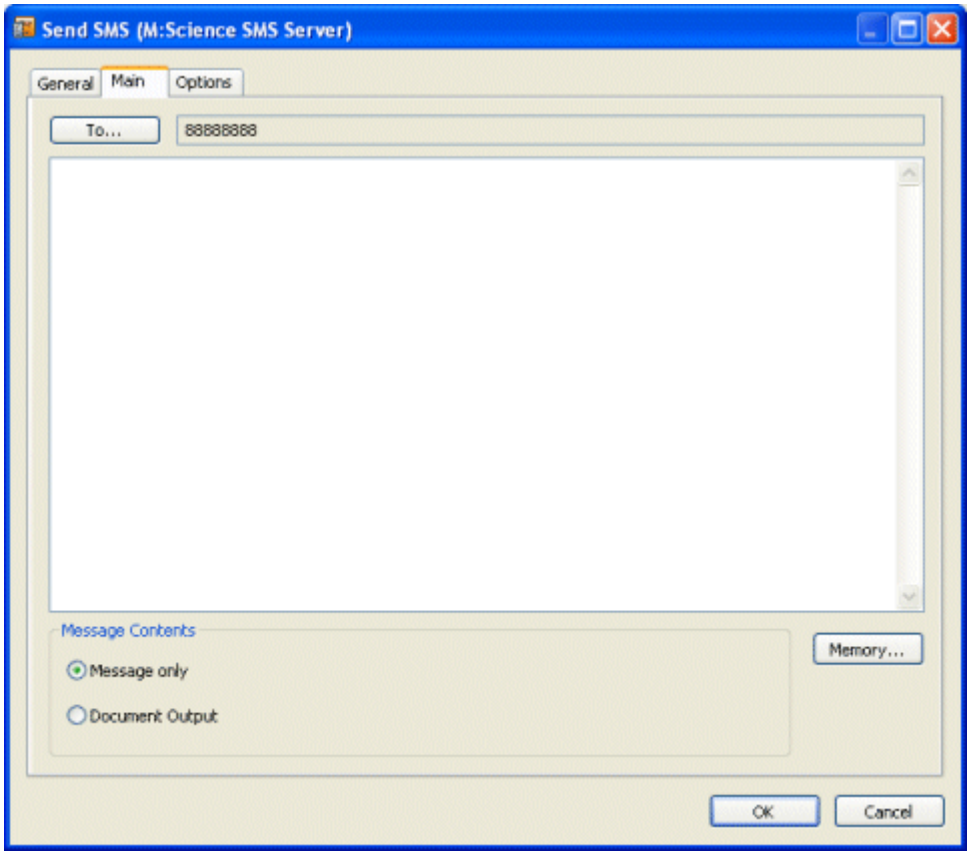


Figure 30 – Send SMS – Main Tab

Features

- Send SMS Messages to any mobile or GSM device
- Static or dynamic messages
- Merge message with data from other actions
- Static or dynamic recipients
- Multiple recipients
- Memory features to memorise Task data

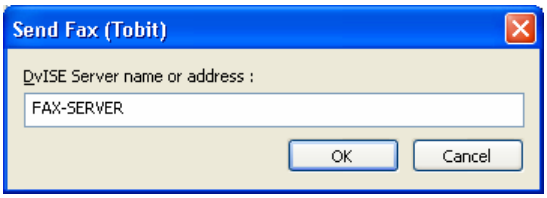
Send Fax (Tobit)

Summary

The Send Fax (Tobit) Tool is used to create a Task Step that sends static or dynamic faxes to one or more recipients via the market-leading mid-range fax products from Tobit Software. The Tool supports Tobit FaxWare and Tobit David, the powerful unified messaging server. Tobit address book and fax cover pages are also supported.

The Step incorporates data from Input and Format Steps into fax messages and then sends them to any number of recipients. It supports both Text and HTML data from Steps such as Format as Text, Format as HTML and Run Crystal Report. If for example, the original Database Query (ODBC) Step used by the Format Step includes a fax number, then this may be used as a dynamic recipient address.

TaskCentre is configured by simply specifying the host server for the Tobit Fax product as shown here.



The figure below shows the 'Main' tab which provides the facility to manually type a fax message. Alternatively, the message can contain a document source from a Format as Text, Format as HTML or Run Crystal Report Step.

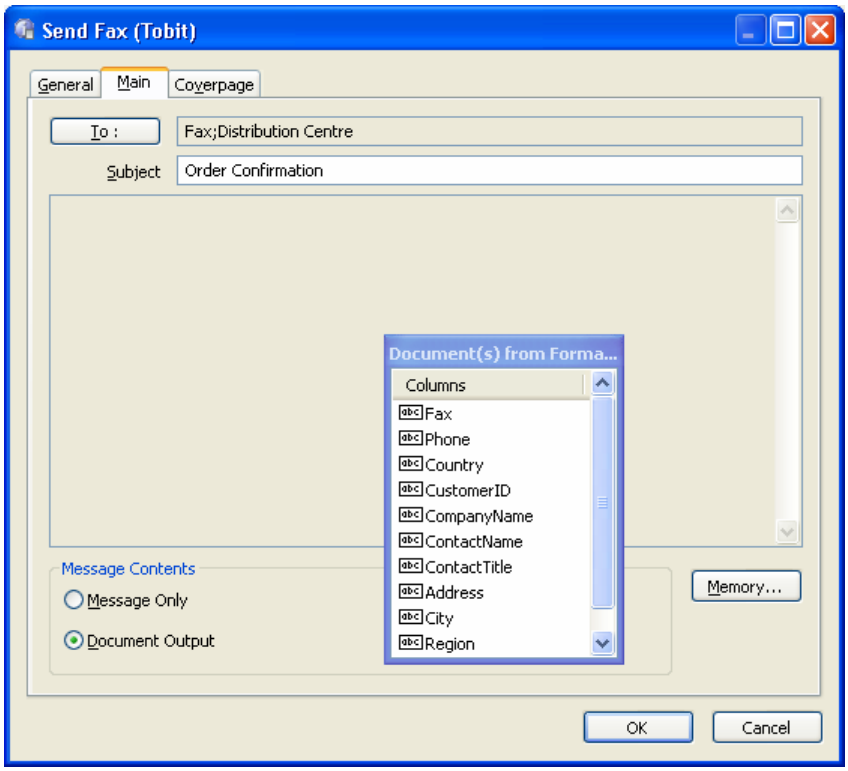


Figure 31 – Send Fax (Tobit) – Main tab

Features

- Send Static or dynamic faxes
- Supports Text and HTML Formats
- Static or dynamic recipients
- Multiple recipients
- Supports Tobit Address Books
- Supports Tobit cover pages
    - Select the cover page per action
    - Static or dynamic cover page variables
- Memory features to memorise Task data

Cover Pages

A cover page can be selected within a Step to be sent with each fax produced. This utilises Tobit software's own cover page feature which provides a number of example cover page templates and the ability to produce customised cover pages. These templates contain variables which may be assigned static values or dynamic values by mapping the variable to the query database fields.

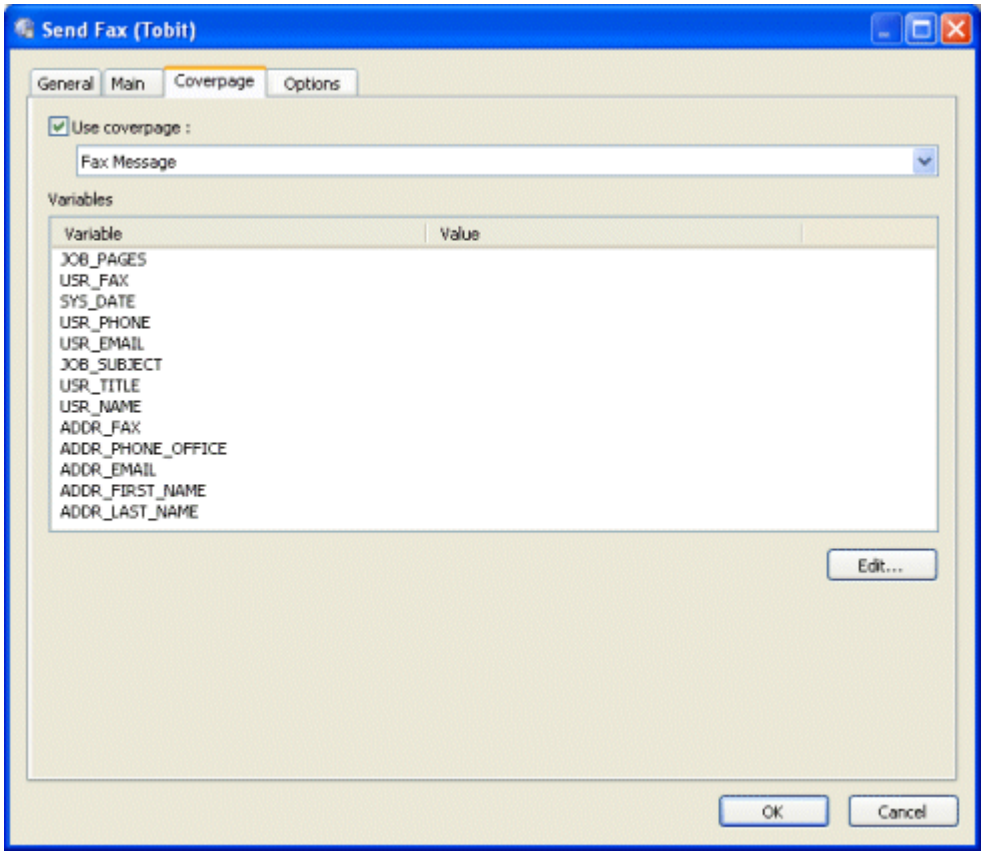


Figure 32 – Send Fax – Cover Page tab

Save As File

Summary

The Save as File Tool is used to create a Task Step that saves the output from a Format type Step as one or more files on disk or to save one or more files from a binary column exposed by an Input Step such as a Database Query (ODBC).

This method could be used to update an Intranet, to save documents for later processing by another Task or simply to expose information to another system or application.

For example, HTML output from a Format as HTML Step may be saved as one or more .HTM files, or output from a Format as Text Step may be saved as a .CSV file for use by another application, such as Microsoft Excel.

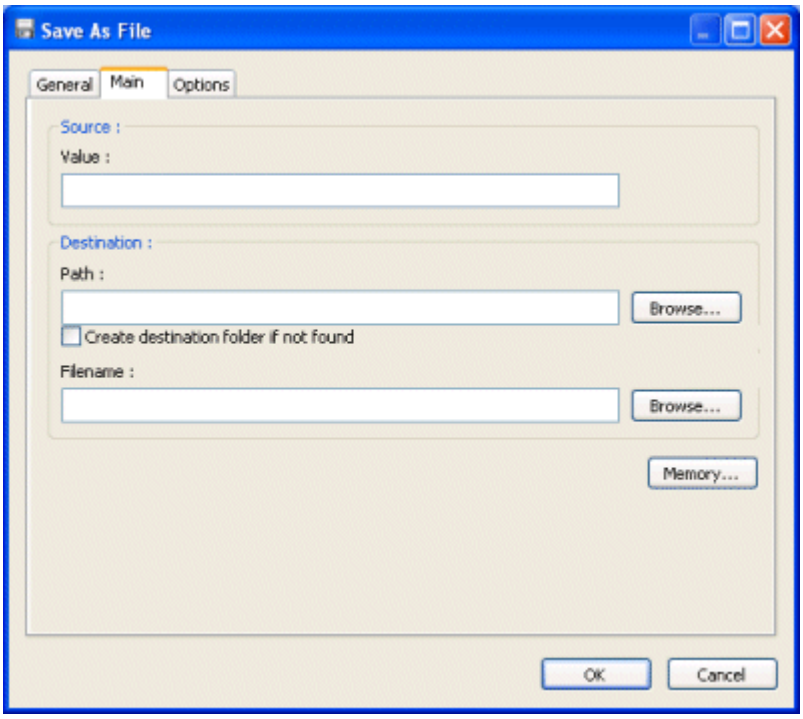


Figure 33 – Save As File – Main tab

Features

- Save single or multiple files to the file system
- Any file type provided by another action
- Binary files from sources such as Database Query (ODBC)
- Static or dynamic paths
- Static or dynamic file names
- Memory features to memorise Task data

Print Document

Summary

The Print Document Tool is used to create a Task Step that provides comprehensive print services for incorporation into any process where a hard-copy output is required. The Print Document Step works by interacting with other Steps that produce textual output such as Format as Text or that expose print capabilities such as Run Crystal Report.

A single Print Document Step can print one or many documents to any printer for which a Windows compatible driver exists.

The Print Document Tool allows the configuration of all the standard print attributes when designing a specific Step but also allows these to be controlled at run-time by using objects from, for example, a Database Query Step.

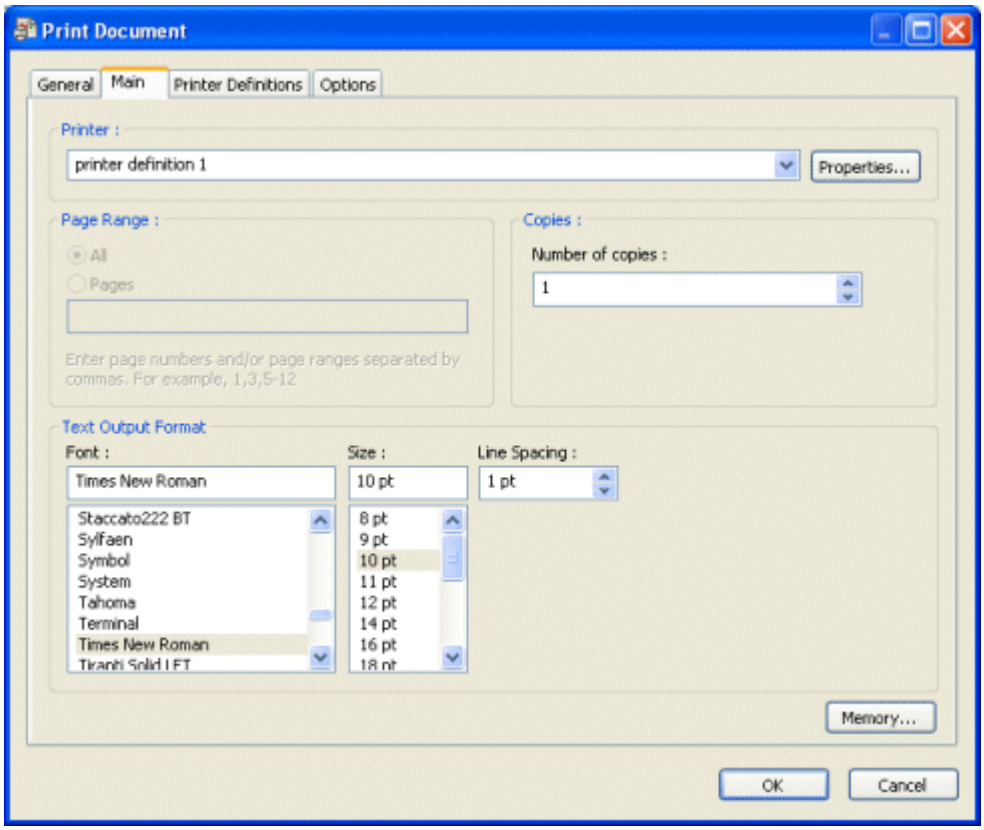


Figure 34 – Print Document – Main tab

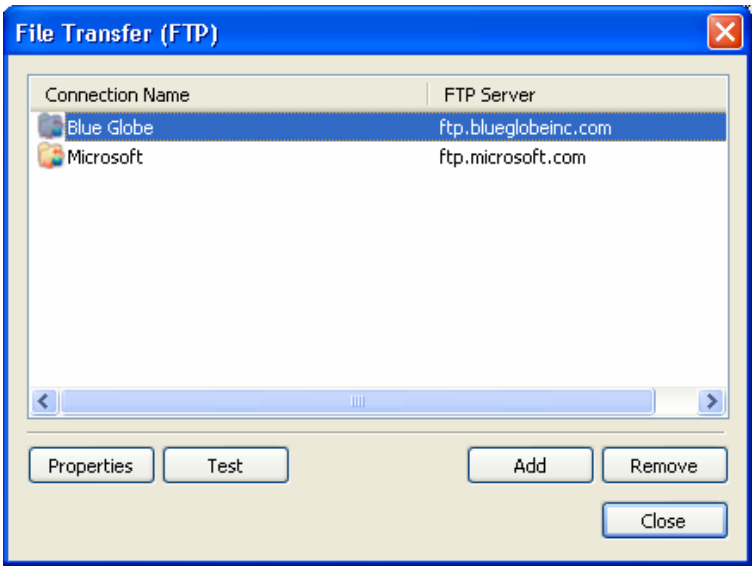
Features

- Print one or multiple document
- Any windows printer driver
- Static or dynamic settings
    - Printer, Paper Size, Paper Source, Orientation, Copies, Collation, Scaling, Quality, Font, Size, Style.

File Transfer (FTP)

Summary

The File Transfer (FTP) Step is used to create a task Step that can upload or download one or more files to and from one or more FTP Servers. When uploading, the files in question may be created by TaskCentre Steps or may already exist within an accessible file system. For example, documents produced by the Format type Steps can be produced dynamically and uploaded to an FTP site in a single Task.



Multiple FTP servers and their authentication details can be defined globally and their use is then transparent in the rest of the product as shown here.

Within each file transfer definition, it is possible to specify whether it is an upload or download, a static or dynamic FTP Server to use including dynamic authentication details plus the details of the files themselves, again static or dynamic.

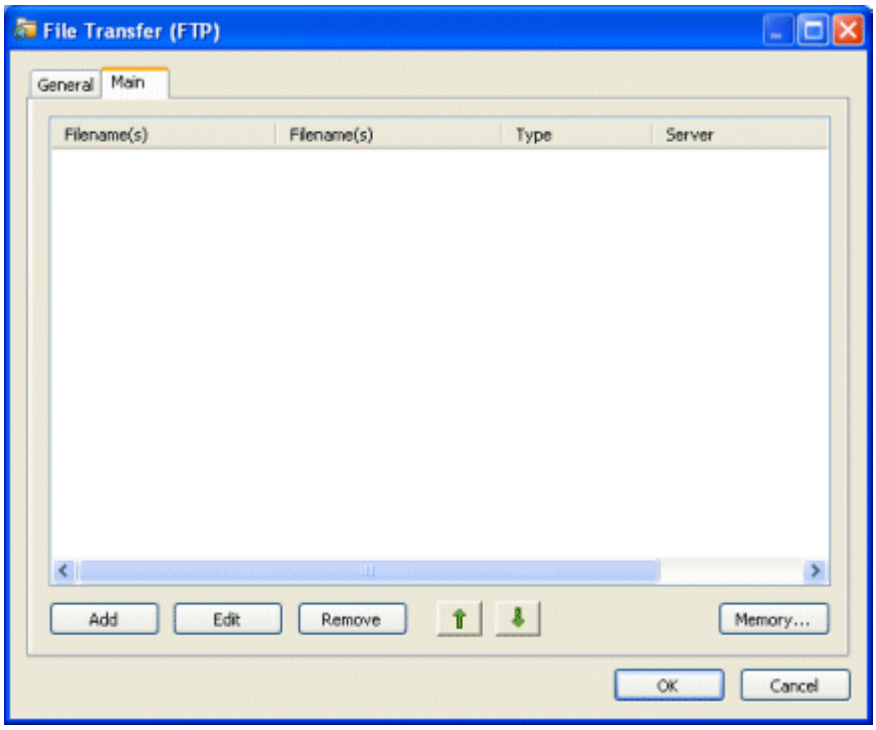


Figure 35 – File Transfer (FTP) – Main tab

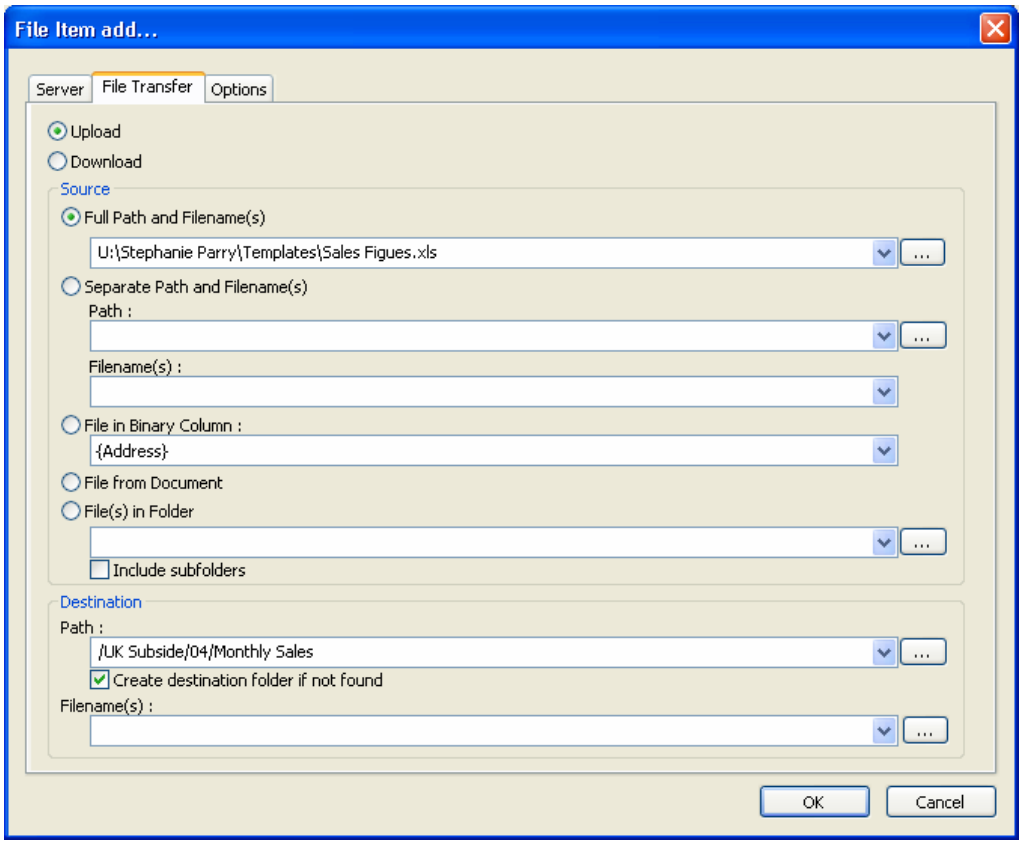


Figure 36 – File Item Add Dialog

Features

- Transfer single or multiple files to and from FTP sites
- Multiple up/down transfer definitions per Step
- Static or dynamic files and documents
- Supports Data in Binary columns from data sources
- Static or dynamic FTP Sites, Ports, Authentication, Passive/Active
- Passive or Active mode
- Supports Binary Transfers
- File exists options
- FTP Site Authentication
- Memory features to memorise Task data created at run time

Execute Tools

Run Program

Summary

The Run Program Tool is used to create Task Steps that run self-terminating shell executable programs including .exe or .bat for example, or in fact any file type that is associated with such a shell executable program. Programs can either be existing programs that exhibit the right attributes or programs specially created to provide integration with proprietary protocols, systems or applications.

As well as the flexibility to create a program that performs virtually any operation, the behaviour of the program can also be controlled dynamically at run time. To enable this, the Step can pass static or dynamic command-line parameters to the program when run. The program can be run once or multiple times, depending on the way the Step is configured.

The results from an Input Step such as the Database Query Step or documents from a Formatting Step such as an HTML or Textual document may be used.

Figure 37 below shows the main Run Program configuration dialog where the executable program is selected and the optional list of command line parameters is defined. Dynamic parameter values can be incorporated from objects from another action such as Database Query (ODBC) by simply dragging and dropping them into the 'Value' column from the pop up dialog.

In addition, documents from a Format type Step can be passed to the program to be manipulated or transferred to a proprietary system, protocol or application.

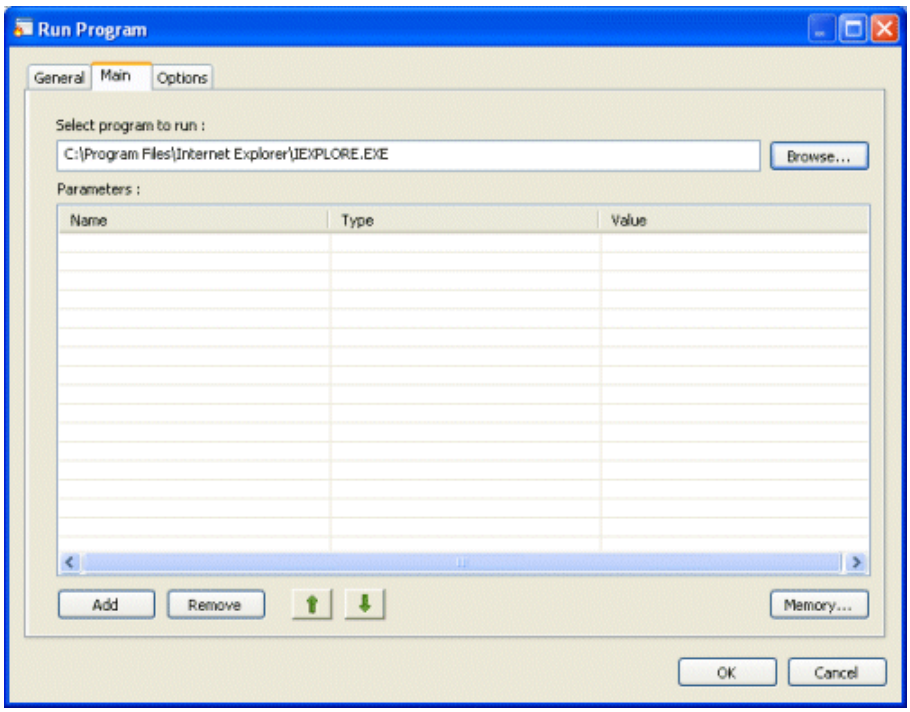


Figure 37 – Run Program – Main tab

Features

- Run any self-terminating shell executable (.exe, .bat, .com or etc)
- ‘Run’ files associated with self-terminating shell executables
- Pass static or dynamic command-line parameters
- Utilise source data from other actions
- Execution failure options
- Memory features to memorise Task data created at run time

Call COM Object

Summary

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

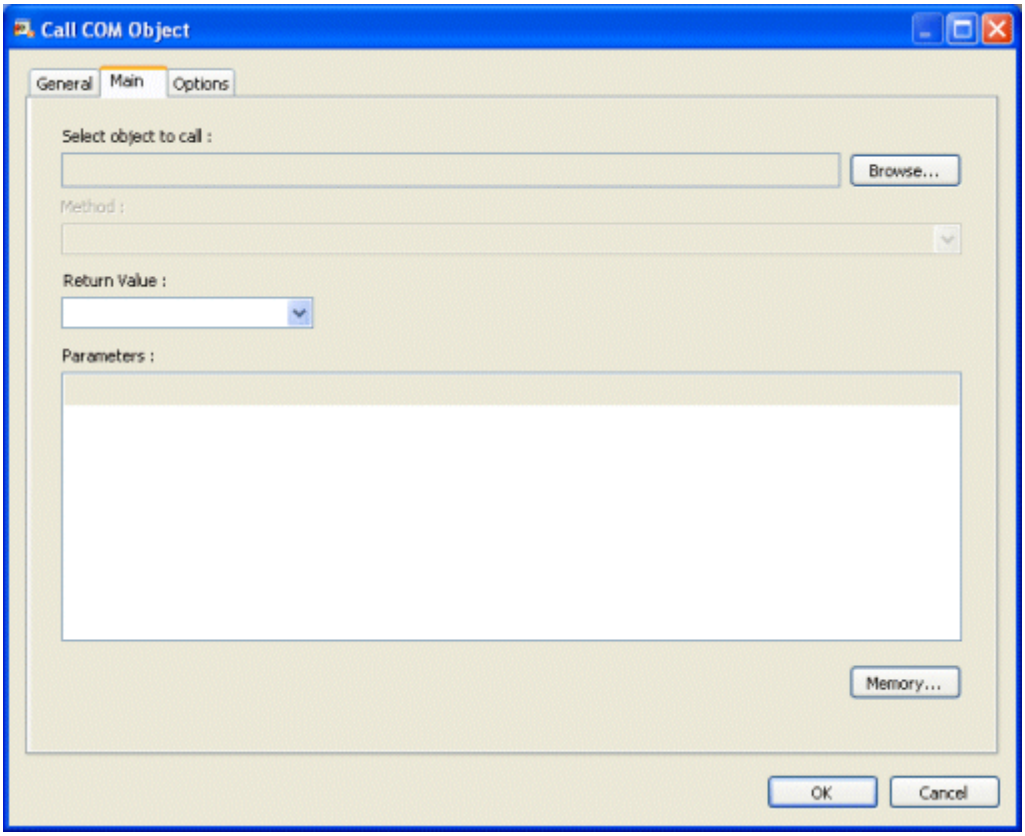


Figure 38 – Call COM Object – Main tab

Features

- Call COM objects one or multiple times
- COM Server (.EXE) or COM .DLL
- Single or multi-use COM Objects
- Pass static and/or dynamic data from other Steps
- Pass Recordset and or Document data
- Call failure options
- Memory features to memorise Task data created at run time

Run VB Script

Summary

The Run VB Script Tool is used to create a Task Step that runs a VB Script file, utilising data produced by other Steps within a Task.

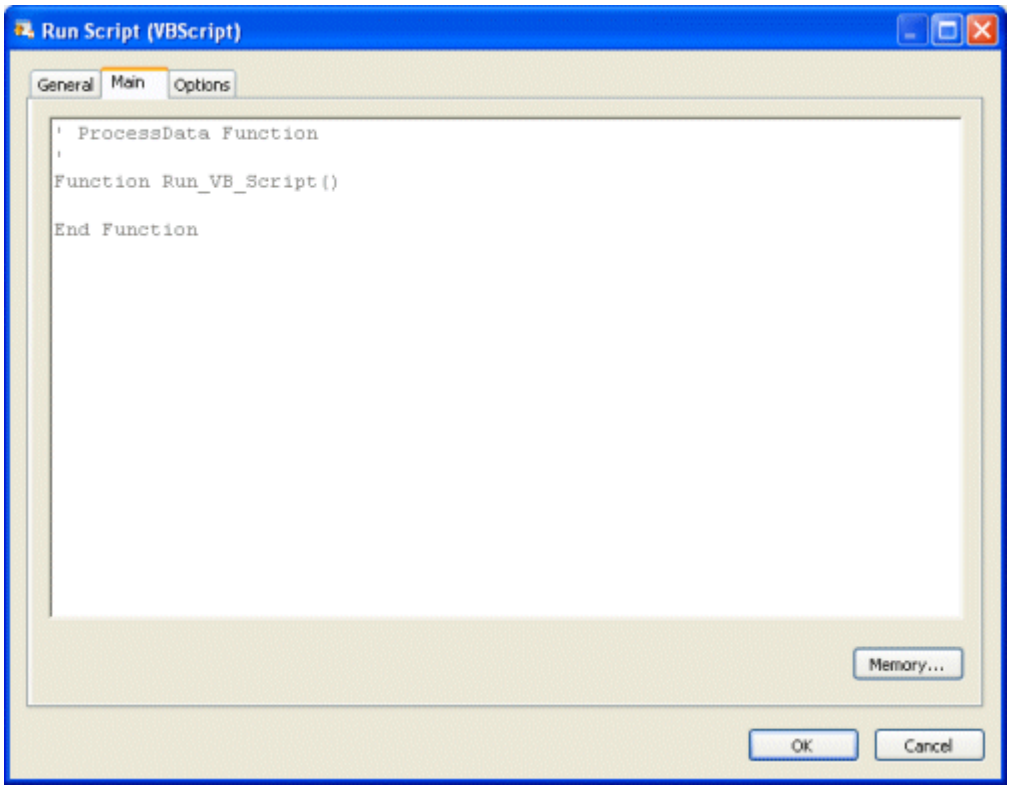


Figure 39 – Run VB Script – Main tab

Features

- Copy and paste script into the editor from external documents.
- Drag and drop common scripts into the script from the Reference tab of the Task Browser dialog.
- Memory feature to memorise Task data created at run time.

General Tools

Decision

Summary

The Decision Tool is used to control the flow of the process based on the result of one or more expressions. The decision works by testing a sequence of Boolean (true/false) conditions in turn until an expression returns 'true' then that branch of the process is followed. For example, a Task could behave differently if it is Friday as opposed to any other day of the week on which the Task runs. For example, Figure 40 below shows a Task which starts with a Decision action, whereby Monday to Thursday a sales team could be given individual daily updates of performance whilst on Friday they are sent a weekly summary.

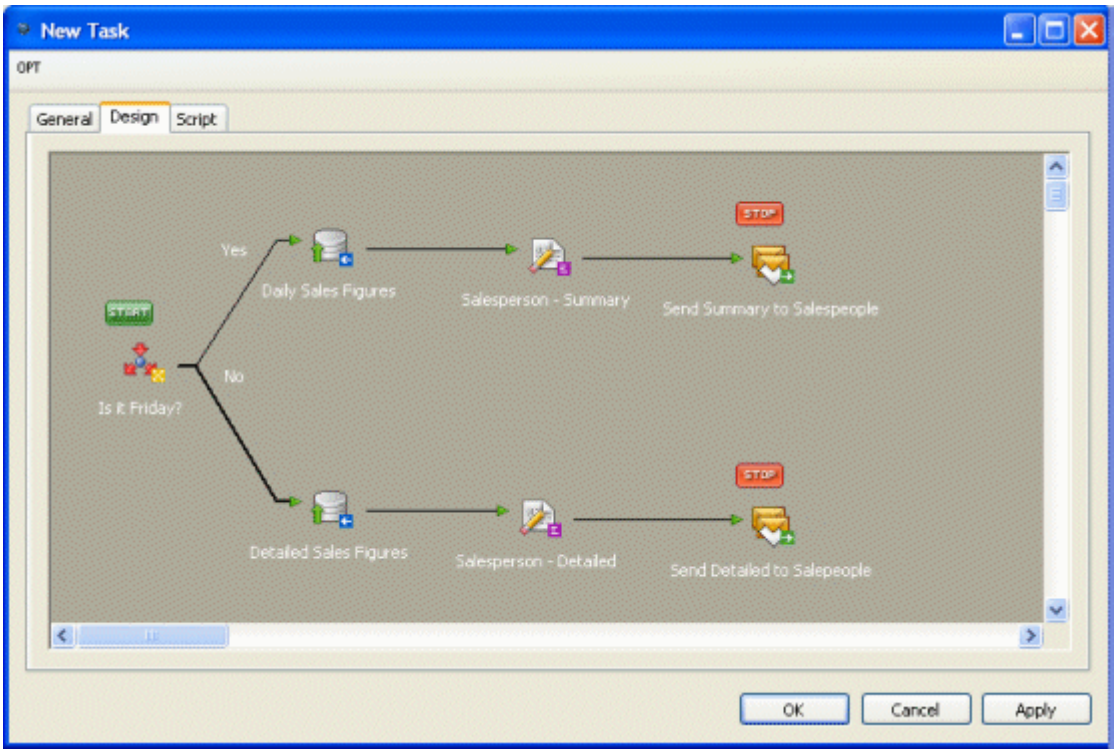


Figure 40 – Task with Decision

Figure 41 below shows the Decision Tool Screen. Pressing the 'Add' button opens the Branch Editor Screen, which enables a new branch to be created or an existing branch to be edited.

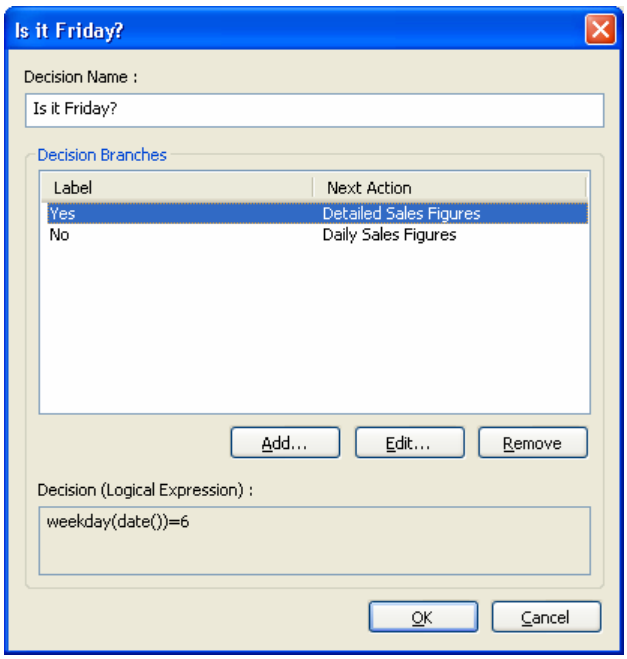


Figure 41 – Decision

Features

- Control the process flow using logical expressions
- Unlimited branches
- IF...THEN...ELSEIF mode
- IF...THEN...ELSE mode
- Native VB Script Expressions
- Full VB Script Function Help
- Visual Branch labelling

Text Parser

Summary

The Text Parser Tool is used to create a Task Step that extracts specific textual data from variables populated by other Steps and then maps the data to other variables for use in further Steps.

Textual data to be parsed is first selected by dragging and dropping a variable containing the data from the Task Browser dialog into the Text Parser configuration dialog. The Step then uses extraction rules to select the textual data to be extracted and mapped to other variables.

These variables may then be used to automatically incorporate the extracted textual data into other Steps.

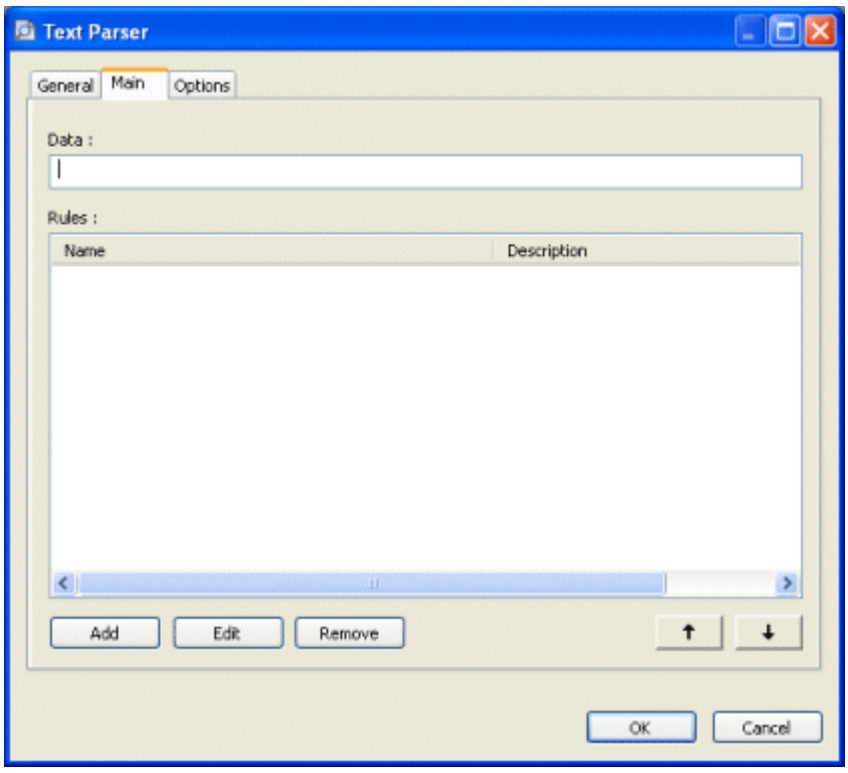


Figure 42 – Text Parser – Main tab

Features

- Single or multiple extraction rules
- Map any available data to multiple variables
- Merge textual data dynamically into other Steps

Data Filter

Summary

The Data Filter Tool is used to create a Task Step that extracts a subset of data from a recordset produced by a Database Query (ODBC) or Call Procedure (OLEDB) Step and provides the data for use in further Steps.

When a Task containing an Output Step that is using the Memory feature is run, a Repository is automatically created for the Task containing the memorised recordset.

When a Data Filter Step is added to the Task, the Repository may be selected and a Repository Filter created within the Step. The Repository Filter is used to compare the new recordset created when the Task is run again, against the existing memorised recordset in the Repository and then set rules to filter the new subset of data produced.

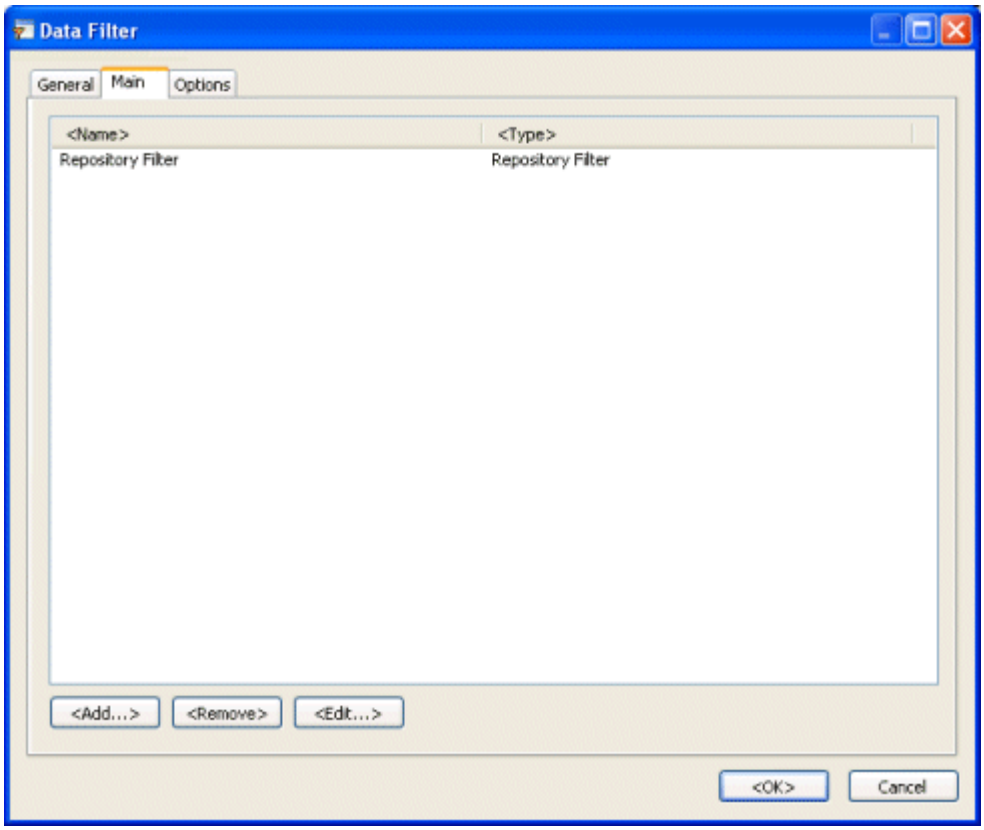


Figure 43 – Data Filter – Main tab

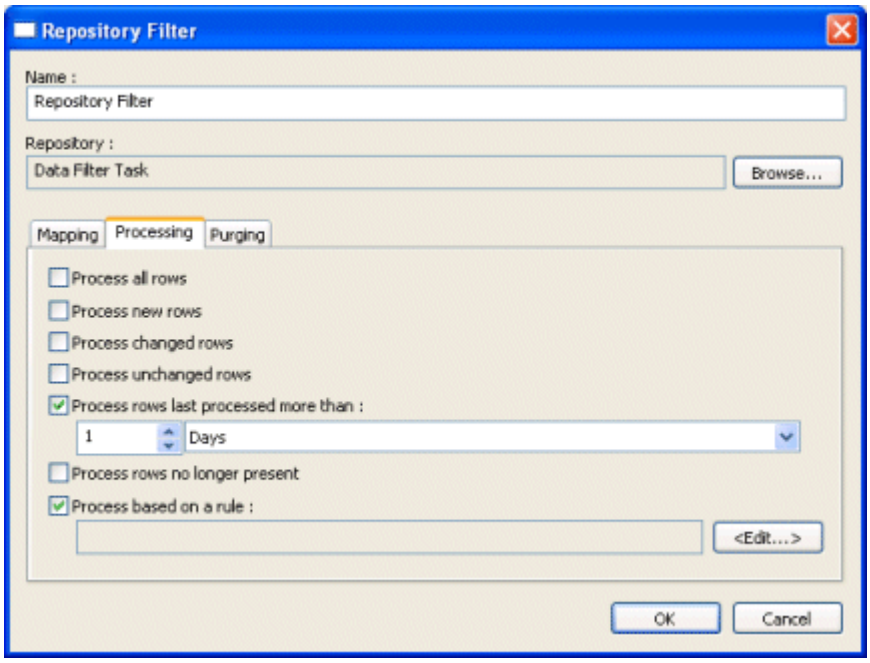


Figure 44 – Repository Filter - Processing tab

Features

- Single or multiple Repository Filters
- Multiple data purging options

Repositories

Summary

When a Task is run, the Output Step can be configured to use the memory feature that enables data for selected columns of the recordset used by the Task to be memorised.

The storage for this memorised data is called a Repository and all Repositories are exposed to the User through the TaskCentre main interface for viewing and editing.

Once a Repository has been created its data and schema may then be edited independently of the Task that created them.

In addition, a Repository may be created independently of a Task. The Repository may then be selected and populated via a Task Output Step, data imported into it from an external source or data manually created.

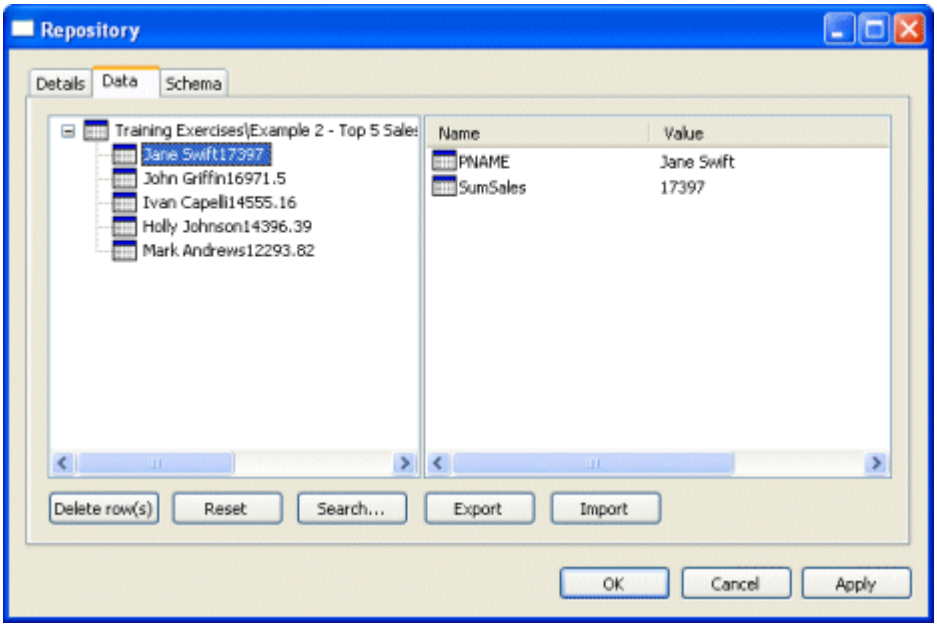


Figure 45 – Repository

Features

The following are just a few examples of how exposing memorised data in a Repository may be used:

- A Repository may be exported and imported as a .xml file which retains all Repository information, including the hierarchical data structure. This is an ideal way to make a backup of the Repository.
- Repository data and schema information may be exported or imported as a .csv file.
- A Repository may be selected for use in any Task so that different columns may be populated from different data sources.

- Repository data values may be manually edited, rows deleted or the Repository 'reset' by deleting all the data.
- The Repository schema may be edited to change column details, insert columns or delete columns to match the data source and even insert child rows within the schema to create a hierarchical data structure.
- Using a Data Filter Step, the current Repository data for a Task may be compared to a new recordset produced at Task run time to filter the data passed to the Output Step for processing.

Managing Tasks

Task Manager

Within the TaskCentre client, the administrator and users (depending on permissions) can create and maintain a user defined hierarchical folder structure. For example, this structure typically mimics the organisations own structure in terms of divisions, departments and teams. Tasks are then created within these folders and are displayed in the pane on the right hand side. See Figure 46 below. Shortcuts to common Folders or Tasks can be created and maintained in the shortcut bar on the Left hand side (not shown below) in flexible groups. A full folder permissions model controls the ability to do this for different users, see Security and Permissions below.

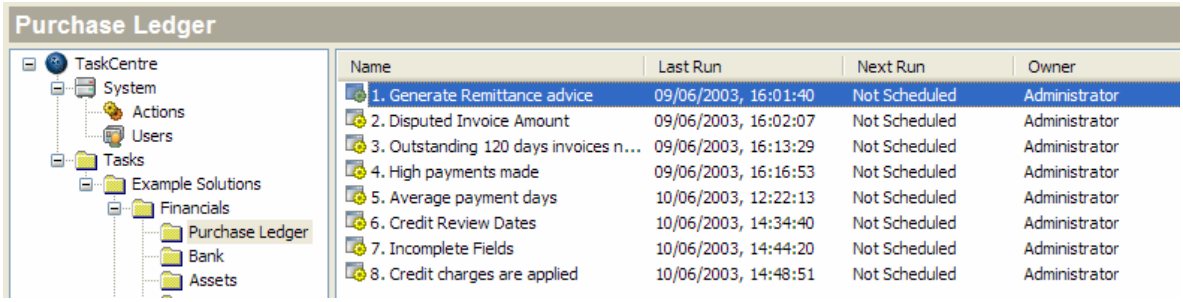


Figure 46 – Sales Folder showing Two Tasks

As you'd expect, Tasks can be created, edited, renamed, deleted, copied and moved within folders and between folders again, depending on the Users permissions.

Exporting and Importing Tasks

Tasks can be easily imported and exported from a TaskCentre implementation to .TKS files. This means that Tasks are fully portable and can easily be moved from one installation to another in this way.

Multiple Tasks and associated folder hierarchy can be exported into a single .TKS file.

Running Tasks

Tasks are typically scheduled to run based on periodic intervals or triggered using an Event Step or some other mechanism via the Task API. But Tasks may also be triggered manually through the client using the 'Schedule Now' feature. This is useful for testing Tasks during the design process and also for Tasks that are sometimes required to be triggered manually for operational purposes.

Security & Permissions

Summary

There are several aspects to TaskCentre security and permissions as described in the following pages. In general, the security and permissions features can be used as little or as much as is required without complicating simpler implementations.

Users

TaskCentre has a built-in Administrator user account that has access to all features and options within the product. Further users can be added by System or Security administrators (See Server Roles below).

Depending on the edition of the product, each user is specified as either using TaskCentre Authentication or Windows Authentication. TaskCentre Authentication requires a username and password whereas when using Windows authentication, it is not necessary to specify logon credentials as TaskCentre obtains these from the Windows session and uses them to authenticate with the server.

Server Roles

There are four Server Roles which serve to categorise administrative access to TaskCentre. The built-in Administrator user account is designated with a Server Role of System Administrator and therefore has full rights to all TaskCentre features. This built-in Administrator user can neither be removed or any of its rights revoked. Initially, only the built-in Administrator can add new users to TaskCentre and set their permissions accordingly. The built-in Administrator may then choose to nominate further System Administrators giving them full control of the system too, or he or she may choose to nominate a Security Administrator to maintain users and their permissions.

In addition to normal user activities such as designing and managing Tasks, the Server Roles give additional administrative privileges to users as follows;

Server Role	Detail
System Administrator	Full access to all features
Security Administrator	Access to User and Tool Access & Permissions
Server Administrator	Access to Global Server options
Tool Administrator	Access to Global Tool options

Tool Access & Permissions

The Tool Access options and associated permissions are specified per user and allow administrators to control which users have access to which Tools. This may depend on whether users have a need to use certain Tools or indeed their knowledge and understanding of the technologies or systems involved with each Tool.

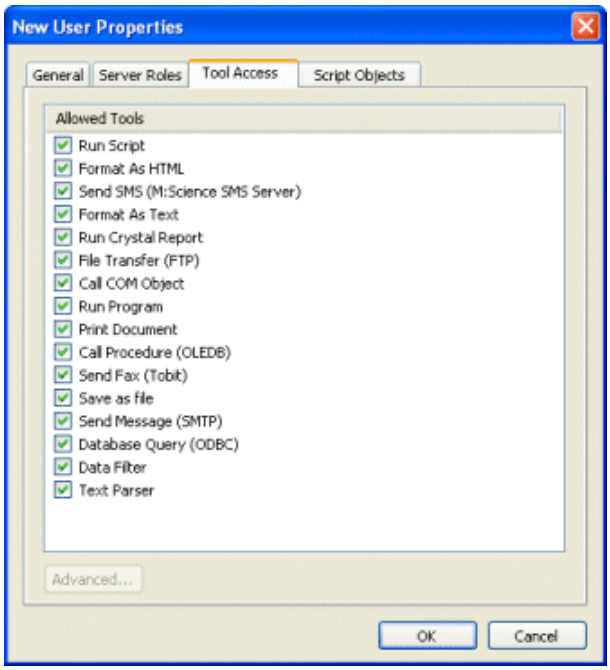


Figure 47 – New User Properties – Tool Access tab

In addition to specifying whether a Tool can be used or not, the Tool may implement Advanced Permissions and options which are also managed here by user.

For example, when using the Database Query (ODBC) Tool, the Administrator may wish to specify which Database Connections a user can utilise depending on their role within the organisation and set their specific credentials for that database.

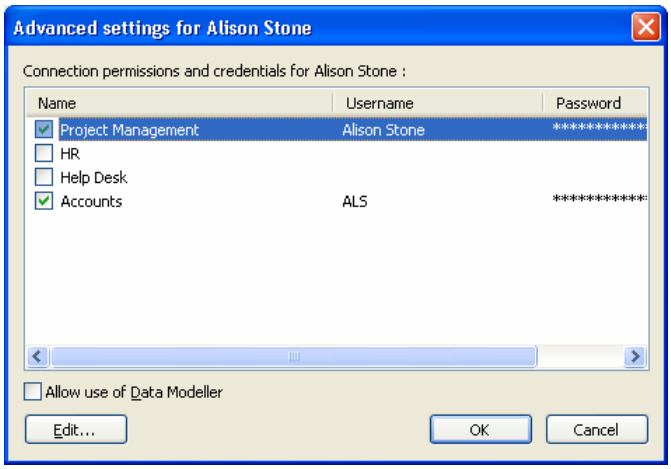


Figure 48 – Advanced Settings for 'User' dialog

Folder Permissions

The folder permission model provides control over user access and activity within individual folders at all levels in the hierarchy. See Figure 49 below. By default everyone has full control of all folders so that folder security need not be considered if not required. Once permissions are altered on a given folder, child folders then inherit these permissions and this process can continue down the tree.

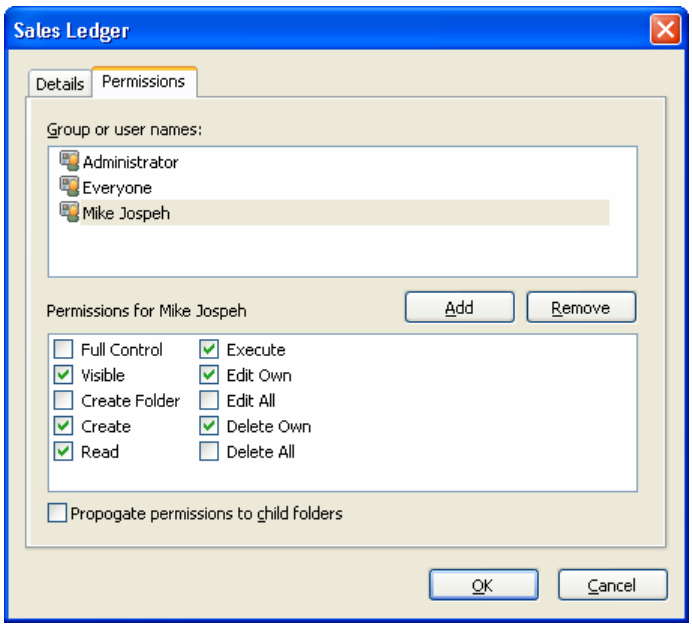


Figure 49 - Permissions tab for the Sales Ledger folder

There are several different permissions that can be denied or granted within a folder to individual users.

Administration

Summary

Due to the nature of the TaskCentre products and their applications, the product provides a range of sophisticated features to administer the product effectively with a minimum of effort.

Crucially, the product is also fully accountable for the Tasks that it undertakes and can be interrogated at will to determine what was done, when was it done and what was the result including any errors or warnings that are generated.

Additionally, the product also provides the ability to generate real-time notifications to Administrators and Task Owners of any anomalies or errors that may occur in the processing of Tasks.

Event Log

The Event log provides complete accountability for all activity within TaskCentre. It is the Administrators most powerful tool in interrogating the system to see the detail of Task processing or other system activity. It can be used simply to confirm that a given Task completed successfully or to determine the cause of an anomaly or failure.

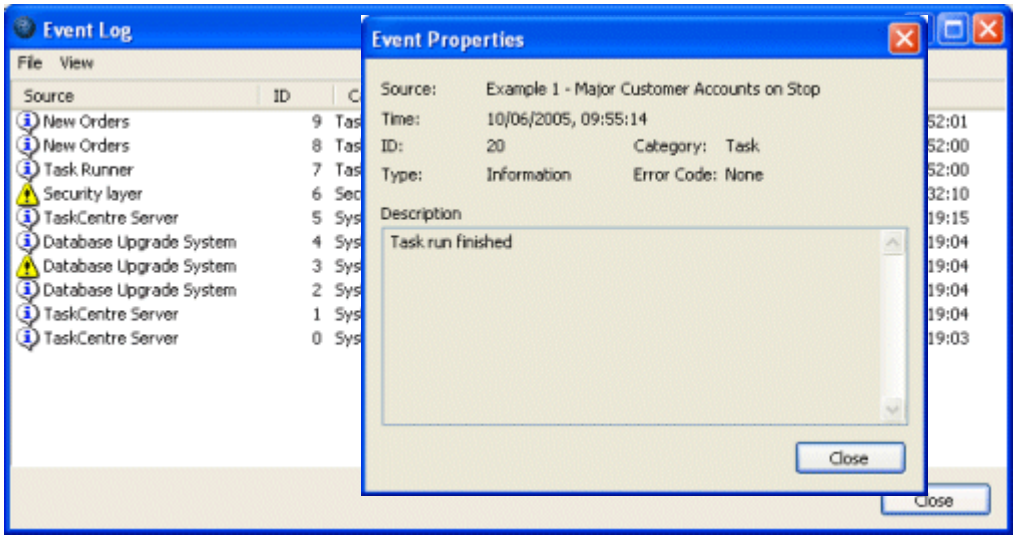


Figure 50 – Event Log & Event Properties dialogs

The event log can be filtered using the properties of the entries or a date range or can be viewed for a specific Task. Subsets of the event log can also be exported very easily to send to support representatives if assistance is required.

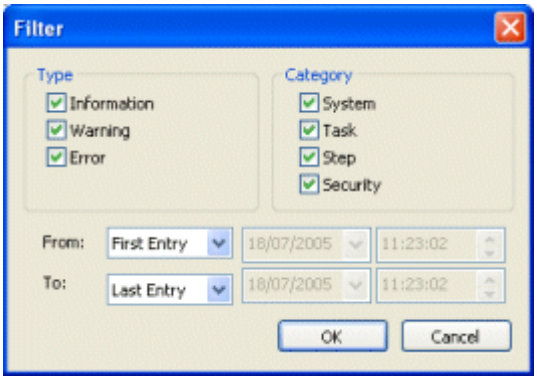


Figure 51 – Filter dialog

Notifications

Notifications provide real-time feedback to administrators and users when an anomaly or error occurs within TaskCentre. The administrator can configure exactly what notifications are required for the different types of potential occurrences. Using these features administrators and users know the instant that something has gone wrong so that they can assess the situation and take remedial action if required.

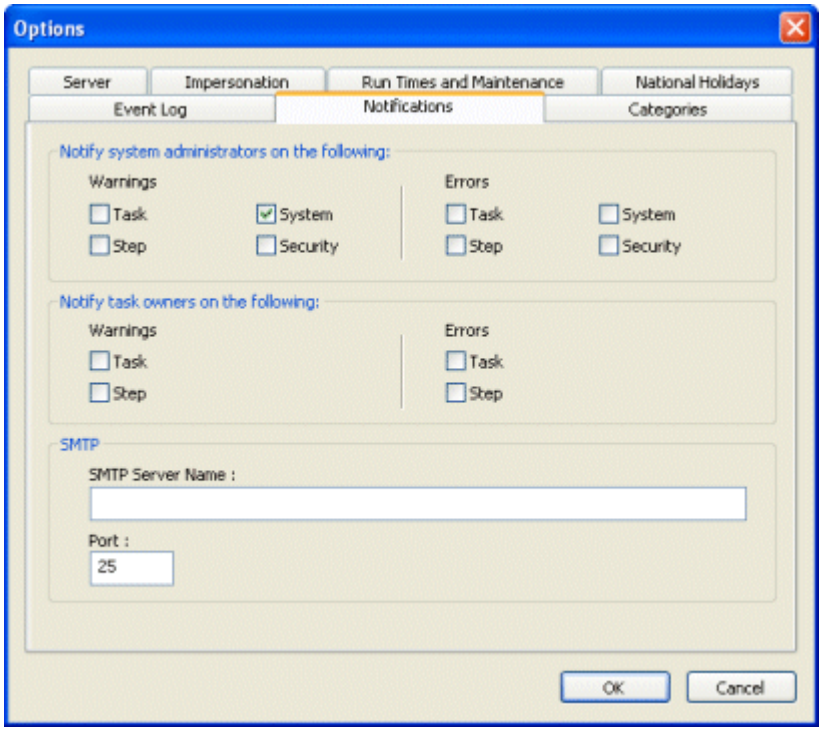


Figure 52 – Options Screen – Notifications Tab

Implementation

Installation

The Installation process for Orbis TaskCentre itself is straightforward on both server and client using a Windows 2000 Installer that automates and requires no complex up-front configuration activity.

Please refer to the document Orbis TaskCentre v4 System Requirements.

Configuration

Due to its very nature, the product does embrace some complex technologies and principles and Orbis therefore advises that the end-user selects a partner from the highly-qualified ORBIS Partner Channel to assist them in implementing the product to their requirements.

Depending on the Tools in use, some prerequisites may exist for one or more Steps and a good knowledge of the technologies, systems or products with which the Steps interact is highly-desirable.

Developer Modules

Data Modeller

Summary

The Data Modeller is a module within the TaskCentre product suite which relates specifically to the Database Query (ODBC) Tool. The Data modeller is used to create Data Models to be used by the Database Query (ODBC) Tool. A Data Model is a semantic or 'user' layer between TaskCentre and an ODBC data source and enables the Database Query (ODBC) Tool to present the data with a structure and use of terminology that is familiar to users.

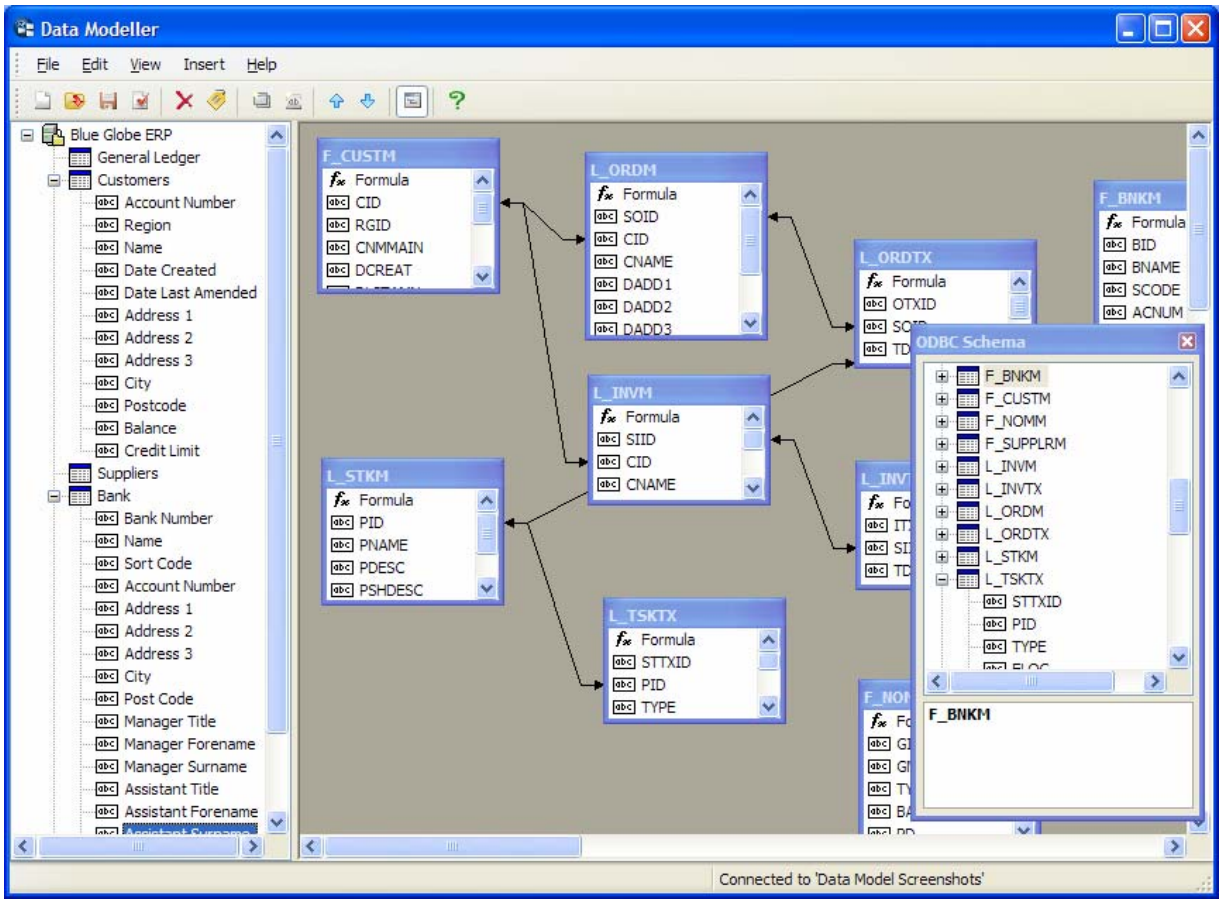


Figure 53 – Data Modeller

Groups and Objects

A Data Model firstly consists of Groups arranged in a hierarchical structure that generally mimic or represent the operational structure of the system that uses the data source. A Group can contain both other Groups and Objects so that a flexible hierarchical structure can be constructed.

Objects represent either column data directly from the source or data that has been manipulated or transformed in some way using expressions. Such expressions may include functions and calculations on data from one or more columns.

Figure 54 below shows the detail behind an Object that provides the credit available to a customer. This information does not exist in the raw data and is therefore a simple calculation between the credit limit and account balance. In addition, a 'Where' clause can be applied to the object so that depending on the context of the object itself, criteria can be automatically applied to any query which contains the object.

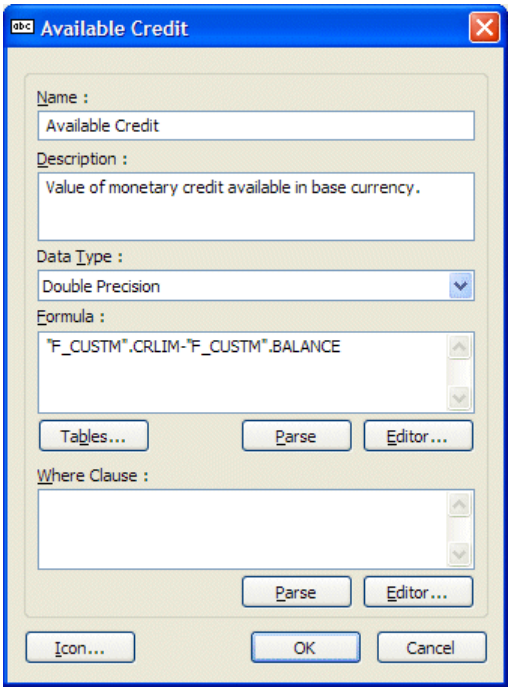


Figure 54 – Object Properties

A formula editor is provided to assist the process of building the formula including function and syntax help.

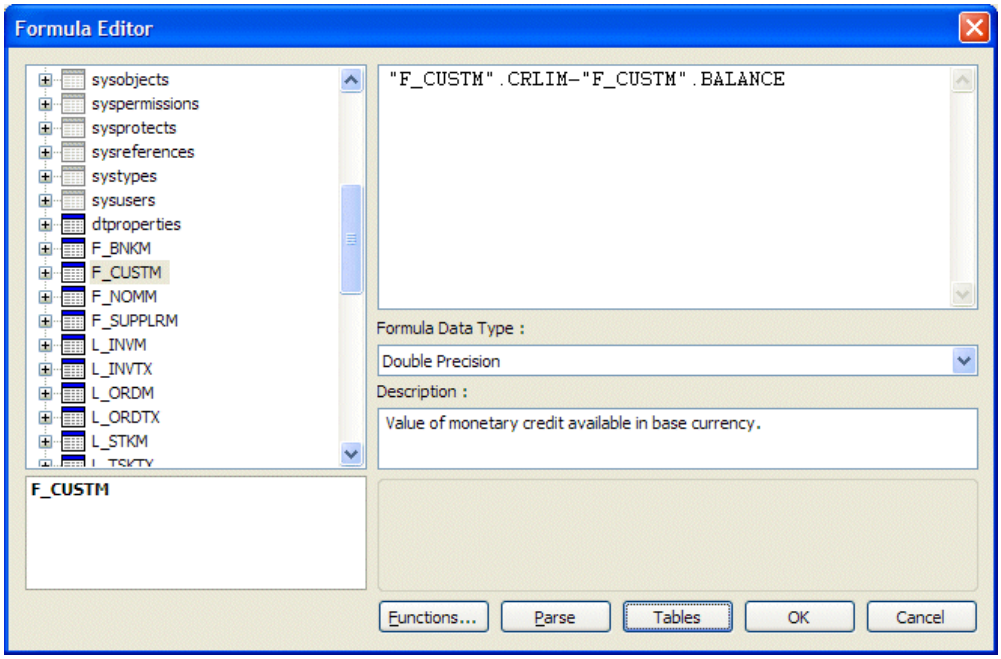


Figure 55 – Formula Editor

Data Relationships

In addition to groups and objects, the Data Model also provides for the definition of data relationships that exist between entities within the data source. In SQL terms, this amounts to defining the 'Joins' that exist between the various tables so that these do not need to be defined again and again when building Database Query (ODBC) Steps within Tasks in TaskCentre.

Task API

The TaskCentre V4 API offers a structured, powerful and easy to use interface to access the TaskCentre Software Suite's Server. It provides the ability to enumerate Tasks in folders, access basic Task information, enable/disable a Task, set/read Task variables, set/read Task categories, set/read Task run parameters and to run a Task.

The API itself is COM based and can therefore be utilised by any development environment supporting COM such as Microsoft Visual Basic, Microsoft Visual C++, ASP or VB Script to name but a few.

Tool SDK

The TaskCentre architecture is such that each Tool is a COM plug-in to the main product. This means that TaskCentre itself is ultimately extensible and that providing the right Tool is available, then virtually anything is possible.

The Tool SDK is a developers toolkit and resource pack to enabled the creation of native TaskCentre Steps.

The Tool SDK includes;

- Visual App Wizard for Microsoft Visual C++
- Step Code samples
- Developers Documentation