

---

## RuleRunner Tasks

Chapter 8 examines the setup and operation of tasks in the *RuleRunner* category.

These are tasks that run primarily in the operating background, as they silently but efficiently apply the processing **rules** that you have previously defined (Chapter 5). The Main and Demo jobs of most applications employ multiple *RuleRunner* tasks.

The Main Job of the pre-configured *1040EZ* application, for example, has three *RuleRunner* tasks: PageID, Recognize and Export. The Demo job has a fourth – a vScan task that runs in response to a single **vScan** RuleSet with just one rule.

Chapter 8 covers these topics:

<b>I. Introduction .....</b>	<b>8-2</b>
<b>II. Setting up a RuleRunner Task.....</b>	<b>8-6</b>
<b>A. RuleRunner Task Setup – Phase 1: Task Project.....</b>	<b>8-7</b>
<b>B. RuleRunner Task Setup – Phase 2: Task Module .....</b>	<b>8-11</b>
<b>C. RuleRunner Task Setup – Phase 3: Task Identity .....</b>	<b>8-14</b>
<b>D. RuleRunner Task Setup – Phase 4: Task Criteria .....</b>	<b>8-16</b>
<b>E. RuleRunner Task Setup – Phase 5: Task Settings.....</b>	<b>8-20</b>
<b>III. RuleRunner Tasks – Operations .....</b>	<b>8-22</b>

## Introduction

Chapter 6 described the structure of your application's Workflow Hierarchy- its workflow, jobs and tasks – and reviewed the steps you take to construct a complete hierarchy.

This chapter focuses on the definition and operation of a special set of tasks...the *RuleRunner* tasks. These tasks apply **rules** that determine how a task is to process a batch, its documents and their pages, and data in the fields of a *source* page.

The Export task of the pre-configured *1040EZ* application is a *RuleRunner* task of limited scope (but unlimited value!) The components depicted on the next few pages contribute to this task's construction and its operations:

- ✓ You'll find these components in the *1040EZ Taskmaster Window* and the *1040EZ Rule Manager Window*.
  - To access the application's *Taskmaster Window*, select **Datacap Taskmaster** from the **Programs** options of your Windows Start button. Open the **Applications** folder and the **1040EZ** sub-folder. Double-click on the **1040EZ Client** icon and enter your User ID and Password to open the application's Taskmaster Client (and the *Taskmaster Window*).
  - To access the *1040EZ Rule Manager*, follow the steps above to open the **1040EZ** sub-folder, then double-click on the **1040EZ Rule Manager** icon.

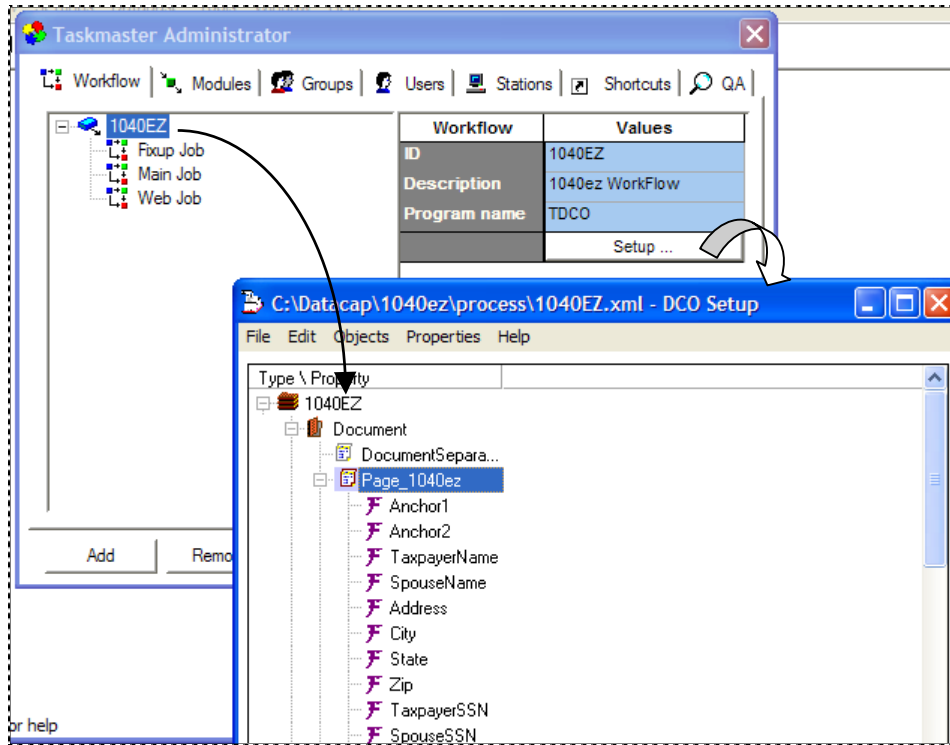
The first example on the next page starts off in the *Workflow* tab of the *1040EZ Taskmaster Administrator* (select **Workflow** from the window's **Settings** file.)

The *1040EZ* workflow is at the top of the **Components** list on the left. Its most important property does not appear until you highlight the Workflow ID and click on the Setup button. This step opens the *Document Hierarchy Setup* window and displays objects at each level of the **1040EZ** Document Hierarchy.

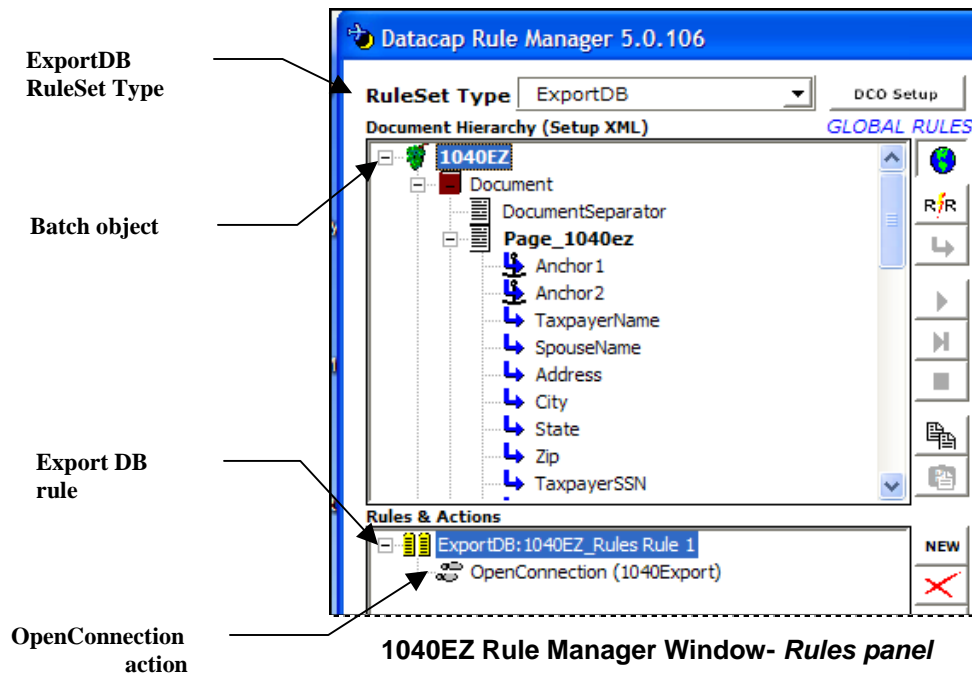
- ✓ **Important!** The definition of a *RuleRunner* task **cannot begin** until the application's Document Hierarchy is firmly in place (Chapter 3).

In the second illustration, the *Rule Manager Window's Rules* panel shows elements of an opening **ExportDB** RuleSet that will help export values to a database. In this example:

- ◆ The **ExportDB** rule is "bound" to the *1040EZ Batch* object at the top of the Document Hierarchy.
- ◆ The single rule consists of a single action: **OpenConnection(DBExport)** opens the application's Export database before a follow-up rule bound to the *Page\_1040EZ Page* object actually transfers values from the Data files in the batch to the Export database.



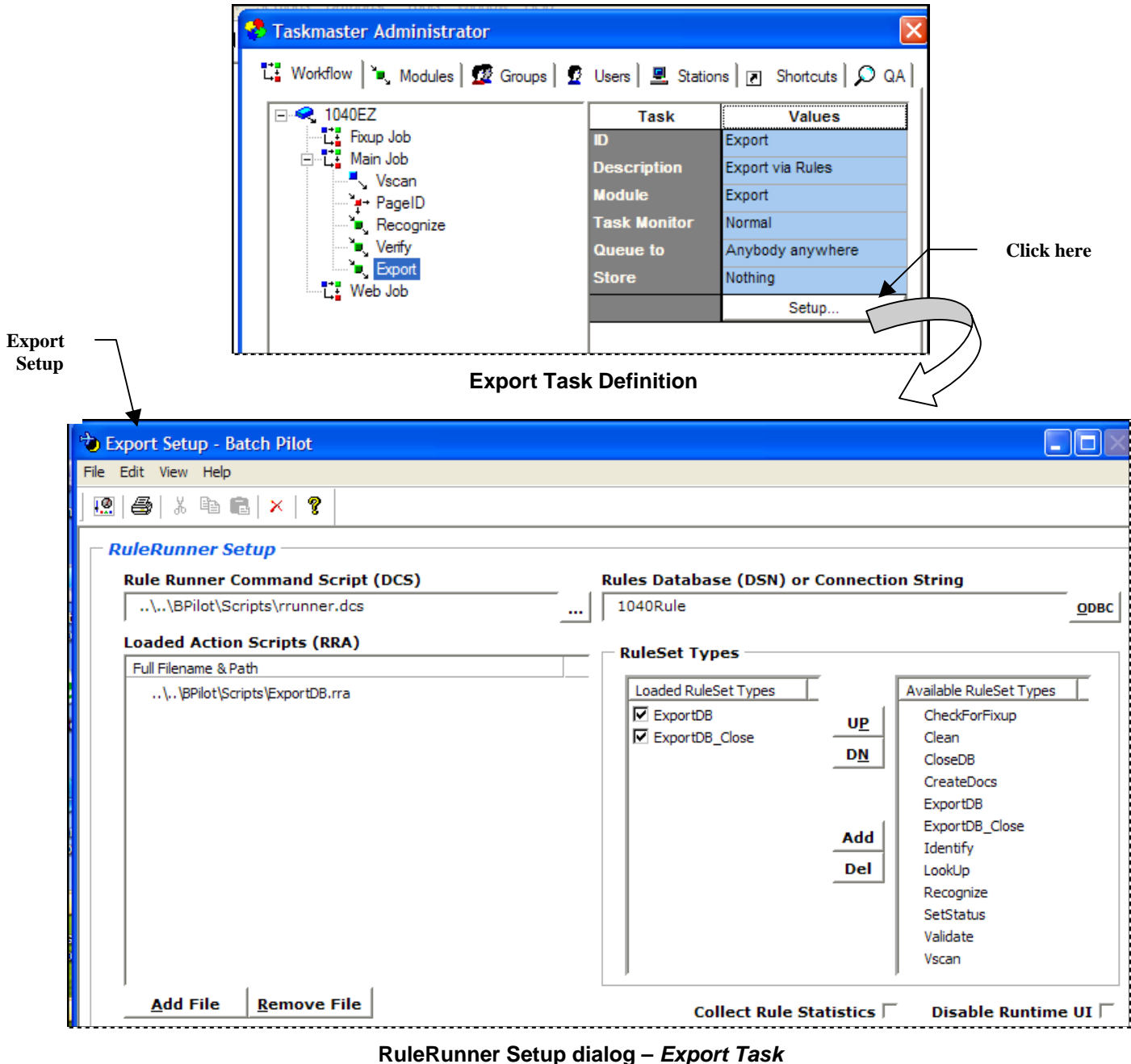
1040EZ Document Hierarchy



1040EZ Rule Manager Window- Rules panel

## Introduction

As soon as the rules have been defined, they can be included in the *setup* phase of the Export Task Definition. Below, if you highlight the *Export* Task ID in the **Components** listings and press the Setup button, the *RuleRunner Setup* dialog will appear on your screen.



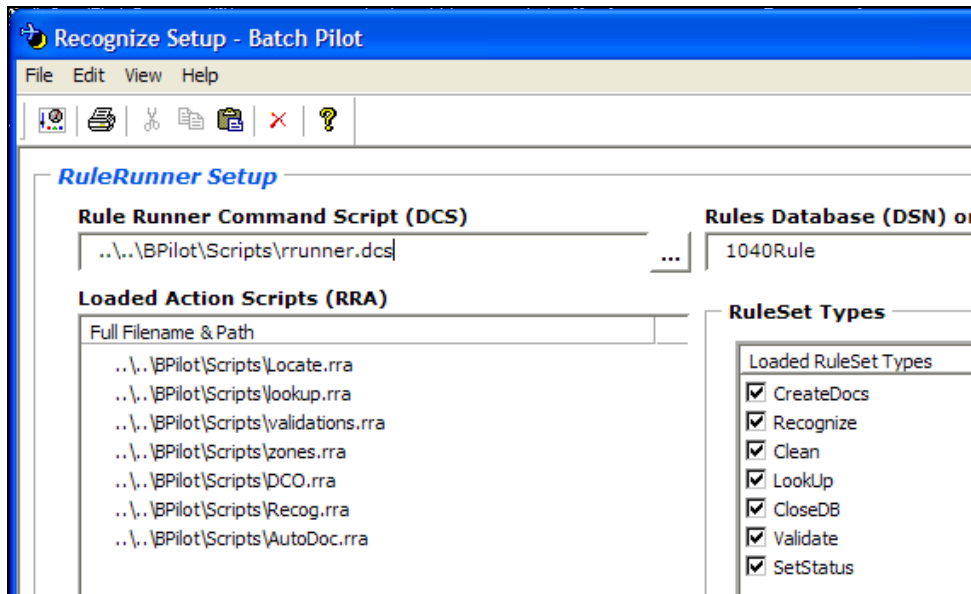
- ✓ Pay particular attention to *Setup* specifications in four fields.
  - ◆ **Available RuleSet Types** lists *all* RuleSet Types containing rules that have been defined and bound to an object of the *1040EZ* Document Hierarchy.

- ◆ **Loaded RuleSet Types** lists those RuleSet Types with rules that the Export task is to apply when it processes a batch - *and* the order in which they are to be applied.
- ◆ **Loaded Action Scripts (.rra)** lists the Actions files (.rra) that supply actions to rules of the RuleSet Types indicated in the **Loaded RuleSet Types** column.
- ◆ **Rules Database or Connection String** contains the Data Source Name (DSN) or Connection String that will link the task to the application’s Rules database. *Important!* This database has information about all aspects of the *Taskmaster Rules* environment you’re working in – including details of the rules you constructed *before* you started work on the current task.

*Important!* Although the examples in this chapter show DSN’s, **Datacap Taskmaster** supports the use of Connection Strings for Microsoft Access, SQL Server and Oracle databases. Appendix A of the *Taskmaster Administrator’s Guide* provides the syntax of these Connection Strings.

On the previous page, the Export task’s limited scope means that it will respond to rules of only two types: **ExportDB** and **ExportClose** – and its rules require use actions from just one Actions file: **ExportDB.rra**. (Chapter 5 explores all aspects of rule composition and definition.)

Below, the Main Job’s Recognize task is a far busier *RuleRunner* task because it applies rules of seven types:

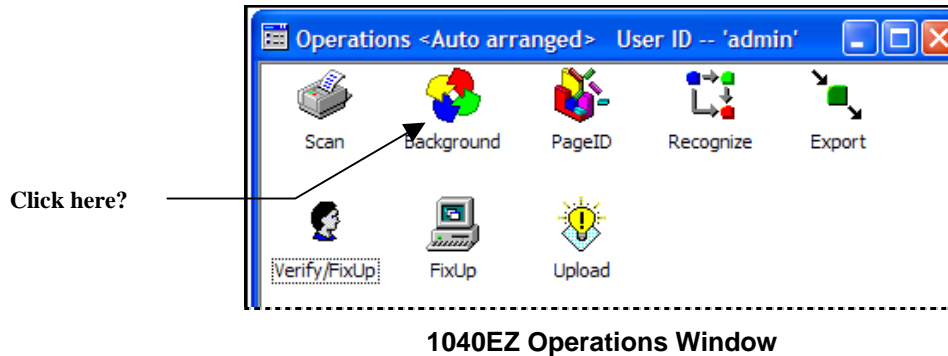


Recognize Task Setup

## Setting up a RuleRunner Task

---

Although the Recognize task has a wider scope, both tasks run in the same background environment (Page 22). Below, a supervisor or operator who handles *RuleRunner* tasks can launch an individual task by double-clicking on its Job-Task shortcut icon - or process batches in the queues of *all RuleRunner* tasks by double-clicking on the **Background** icon.



## Setting up a RuleRunner Task

*Taskmaster* provides standard procedures for setting up a *RuleRunner* task of any kind – a Recognition task or Export task, for example.

However, these procedures are effective only *after* you have provided your application with:

A fully configured Document Hierarchy (Chapter 3)

At least one *zoned* fingerprint (Chapter 4)

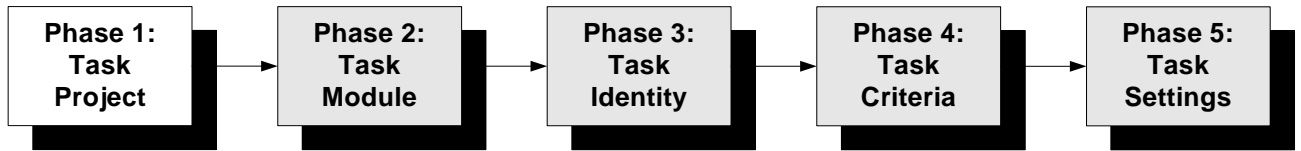
Fully defined rules of the RuleSet Type(s) that the task will apply when it processes a batch (Chapter 5).

A Workflow Hierarchy that includes the job to which the *RuleRunner* task will be assigned (Chapter 6).

- ✓ *Taskmaster* automatically supplies other components such as the task's *setup* and *runtime* forms.
- ✎ When the *Taskmaster* Application Wizard (Chapter 2) sets up an application, it automatically provides the application with skeletal components of various *RuleRunner* tasks – including a Document Hierarchy, Task Projects and Task Modules. Chapter 5 describes the steps you would take to add flesh to these components.

## RuleRunner Task Setup – Phase 1: Task Project

Phase 1 of a *RuleRunner* task's setup defines a **Task Project**, exclusively for the task:

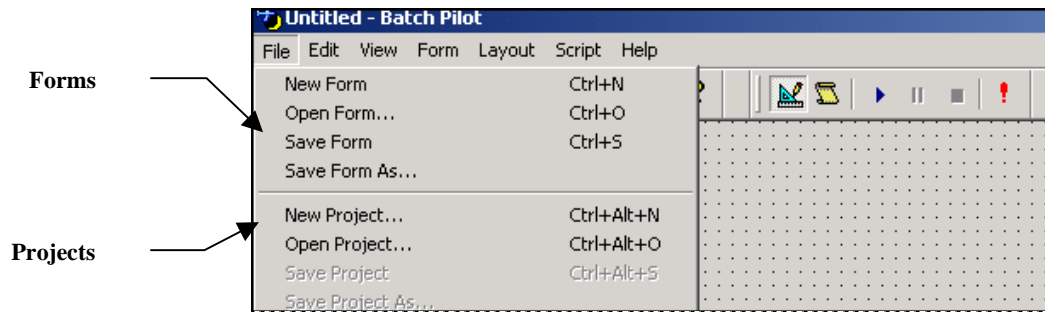


A Task Project is a file – **Export.bpp**, for example – that you construct in the *Batch Pilot* workshop and place in your application's **Process** directory (Chapter 6). The file contains details of the Task Project's connections to:

- ◆ The application's Document Hierarchy.
- ◆ The Workflow Hierarchy.
- ◆ The task's *Setup* parameters.
- ◆ The task's *runtime* dialog.

The Task Project also contains the code that is responsible for much of the task's definition, and for its operations.

- ✓ To access *Batch Pilot*, select **Datacap Taskmaster** from the **Programs** options of your Windows Start button. Open the **Batch Pilot** folder and double-click on the **Batch Pilot** icon to open the *Batch Pilot Window*.



**Batch Pilot Window – File menu**

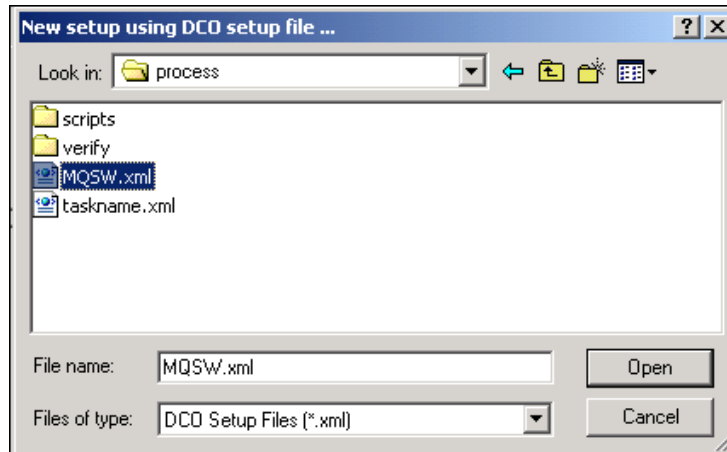
The opening segments of Phase 1 will use the **Project** and **Form** items of the *Batch Pilot Window's* **File** menu. The *Guide to Batch Pilot* – and *Batch Pilot Help* – explain the window's menus and toolbars.

## Setting up a RuleRunner Task

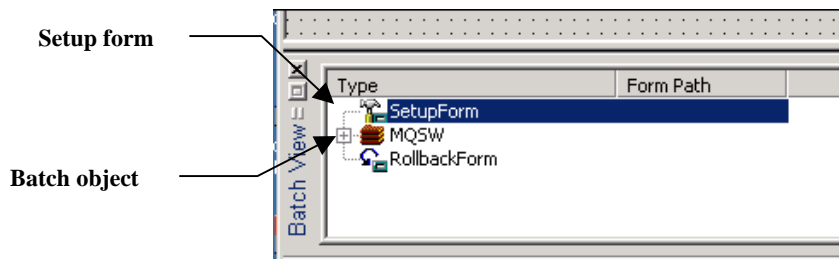
### To Assemble a RuleRunner Task Project

Step	Action
------	--------

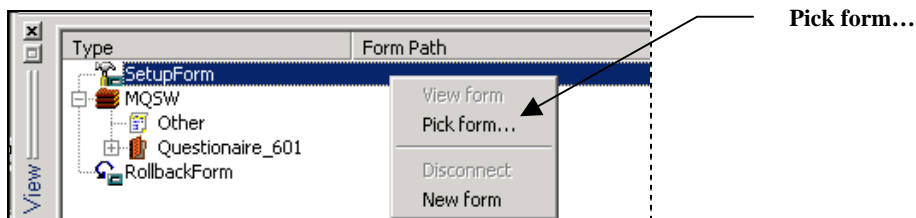
1. Be sure that the **Workflow** component of the application's Workflow Hierarchy includes a Document Hierarchy file (**mqsw.xml**, for example).
2. Select **New Project** from the *Batch Pilot Window's* **File** menu. *Batch Pilot* will instantly ask you to enter the name and path of the Document Hierarchy file in the *Open File* dialog.



3. Select the Document Hierarchy file (.xml) from your application's **Process** directory, and click on the Open button to return to the *Batch Pilot Window*.
4. Confirm that the **Batch View** area at the bottom of the window displays a **Setup** form item in the **Type** column, as well as the **Batch** object of the Document Hierarchy you've specified (*MQSW*, in the example.)



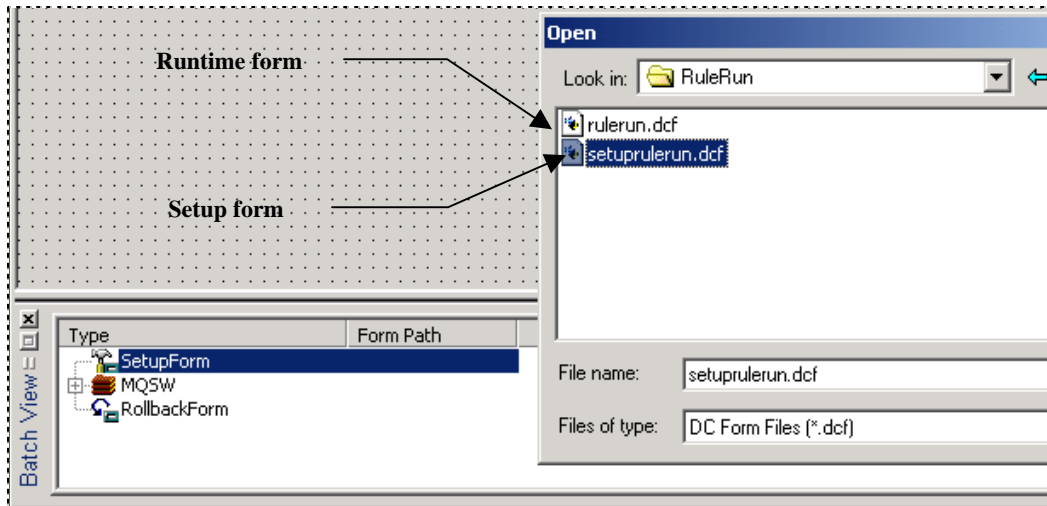
5. Highlight the *SetupForm* listing and right-click in the **FormPath** column. Select the **Pick form...** option.





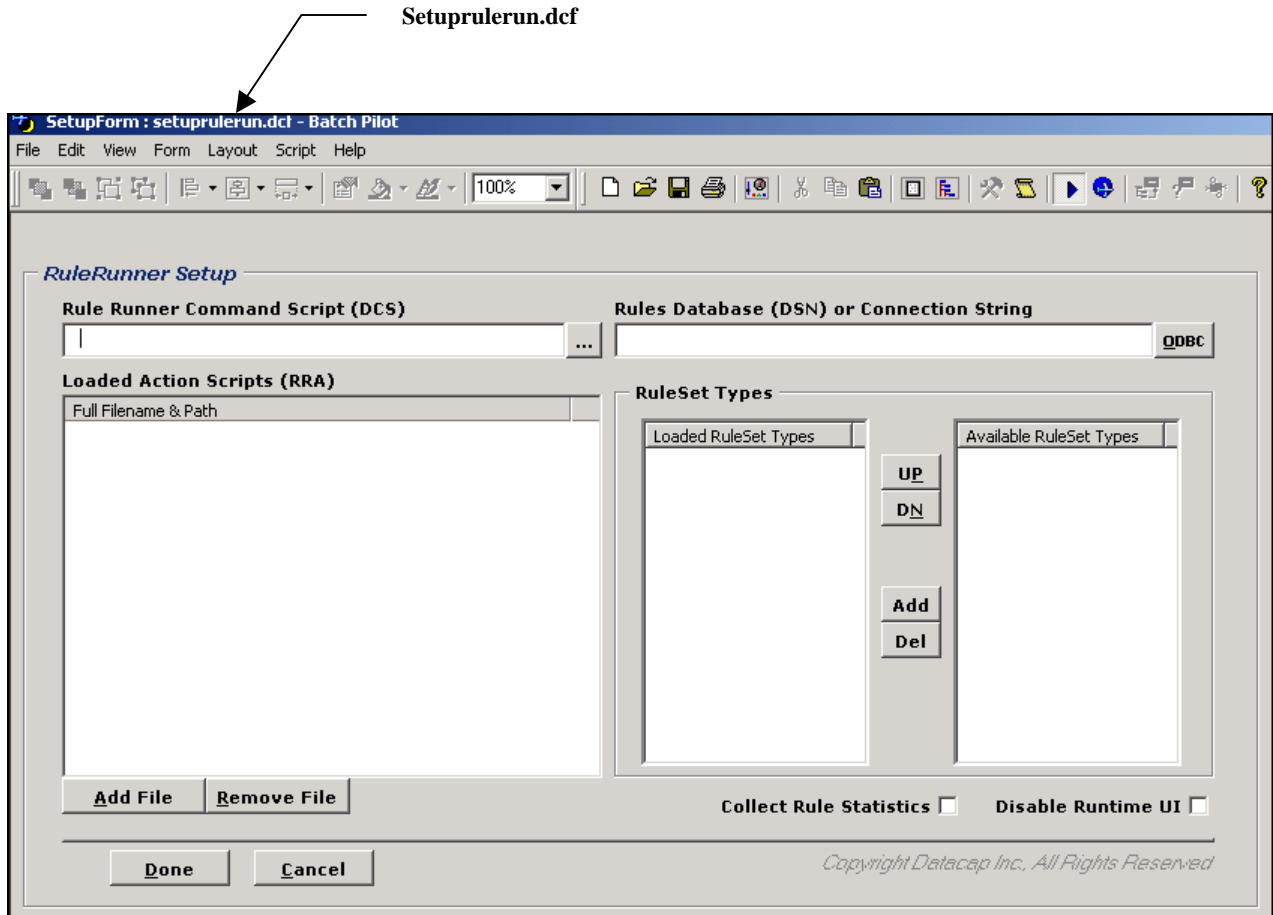
To Assemble RuleRunner Task Project (continued)

- | Step | Action  |
|------|---|
| 6.   | Use the <i>Open File</i> dialog to navigate to the <b>Datacap</b> directory's <b>BPilot</b> folder.                                     |
| 7.   | Click on the <b>Open</b> button.  |
| 8.   | Confirm that the <i>RuleRunner Setup</i> form is now a property of the Task Project (see the illustration of the next page.)            |
| 9.   | Highlight the Document Hierarchy's <b>Batch</b> object in the <b>Batch View</b> area ( <i>MQSW</i> in the examples.)                    |
| 10.  | Right-click on the listing and select <b>Pick Form...</b>   |
| 11.  | Return to the <b>Datacap</b> directory's <b>BPilot</b> folder...and to the <b>RuleRun</b> sub-folder.                                   |
| 12.  | Select <b>RuleRun.dcf</b> as the project's <i>runtime</i> form – the form that will appear on the operator's screen when the task runs. |



13. Select **Save** from the *Batch Pilot Window's File* menu to save this Task Project as **export.bpp** (for this example). This makes the Task Project instantly available to the upcoming Export Task Definition (in this example.)
14. Be sure that the active **Toolbars** of the *Batch Pilot Window's View* include the *Layout* toolbar.
15. Press the **Run Setup** toolbar icon (▶). *Batch Pilot* will display the *Setup* dialog for your *RuleRunner* task (Page 16).

## Setting up a RuleRunner Task

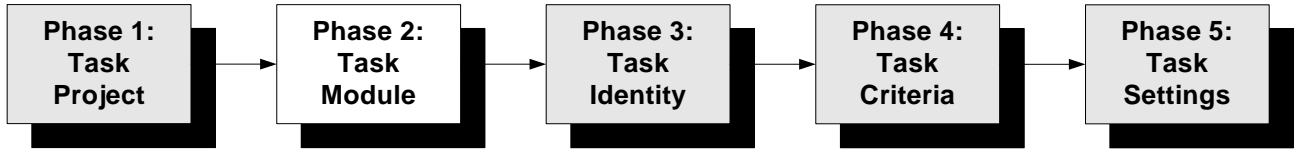


RuleRunner Task Setup dialog

- ✓ The *Setup* dialog above is empty because it is a component of a *RuleRunner Task Project* rather than a contributor to a *RuleRunner Task Definition*. Later, when you define the task, this dialog will be filled with essential information.

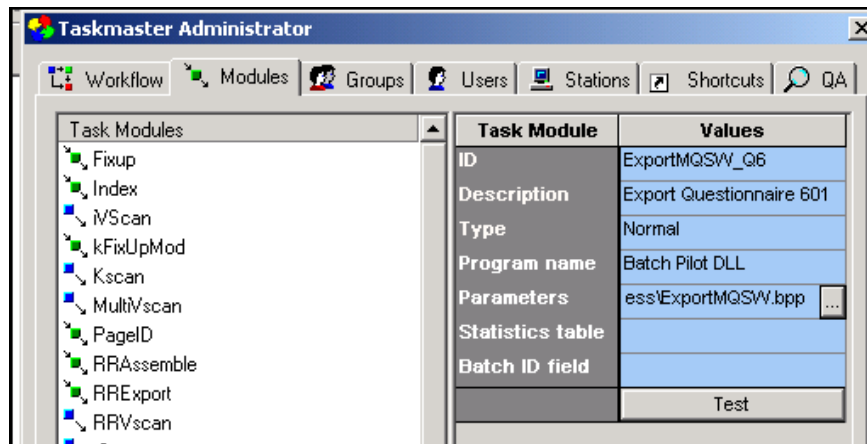
## RuleRunner Task Setup – Phase 2: Task Module

This phase defines the **Task Module** that will connect the *RuleRunner* task you'll define in the remaining phases to the **Task Project** you put together in Phase 1.



To define the Task Module, take the steps below. **Remember!** You cannot define a Task Module until a Task Project is firmly in place (Page 7).

Step	Action
1.	Open the <i>Modules</i> tab of your application's <i>Taskmaster Administrator</i> .
2.	Click on the Add button to clear the fields in the <b>Values</b> area on the right.
3.	Enter a unique Module <b>ID</b> and a brief but important <b>Description</b> of the module.



4. Select *Job Router* from the **Type** drop-down list *if* this *RuleRunner* task can divert batches to a *child* job for review and repair when the task confronts unusual processing conditions (Chapter 10). Otherwise, select *Normal*.

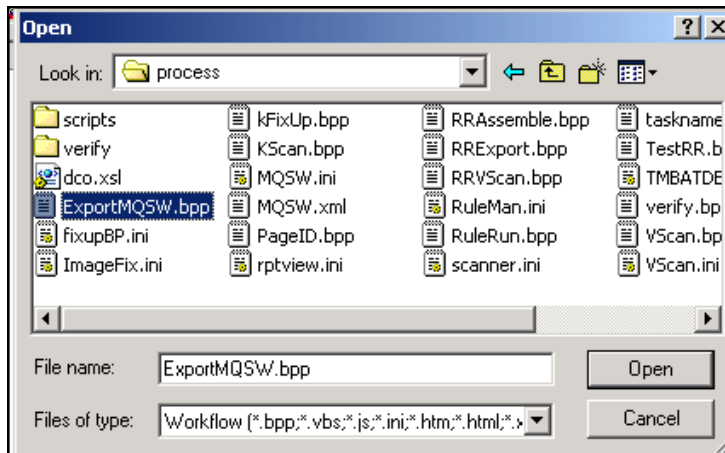
A *RuleRunner* task that applies **Recognition** rules may well branch “problem” batches to a FixUp job. In such a case, its module’s **Type** value would be *Job Router*.

5. Select *Batch Pilot DLL* from the **Program Name** drop-down list.

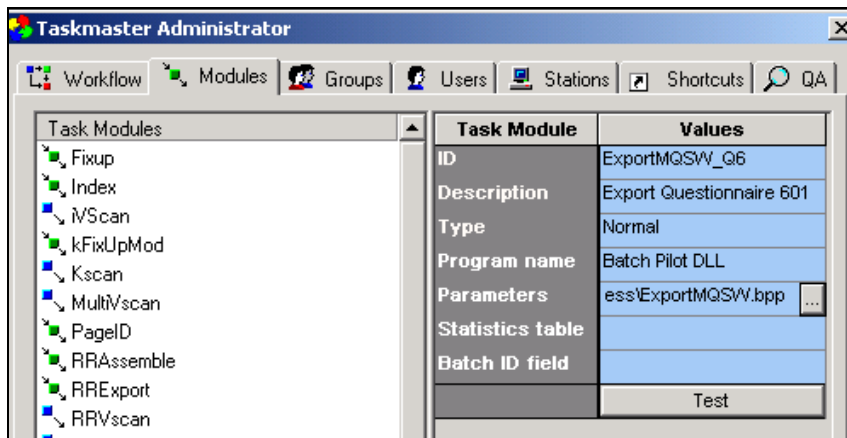
### To Define a RuleRunner Task Module (continued)

Step	Action
------	--------

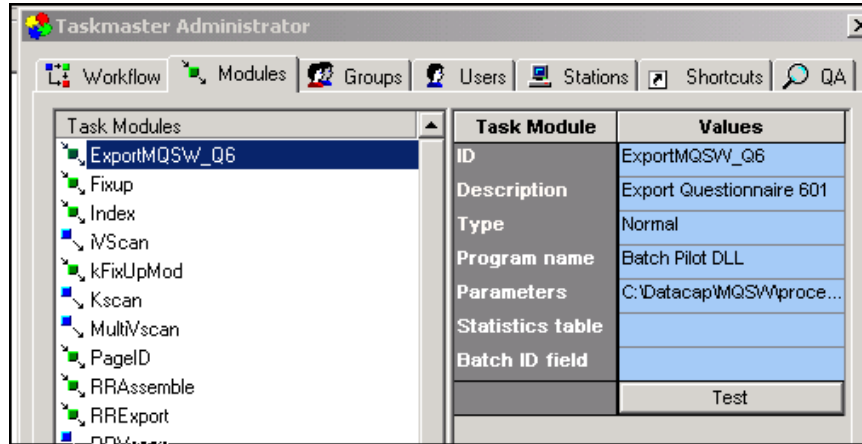
- Click once in the **Parameters** field to display the field's Browse button.
- Click on the Browse button to retrieve the *Open File* dialog. Select the Task Project file (.bpp) you assembled in Phase 1 (Page 7).



- Press the Open button to enter the file's name and path in the **Parameters** field.



- Press the Apply button at the bottom of the *Taskmaster Administrator*. Confirm that the new module's ID is now part of the **Task Modules** list on the left-hand side (illustrated on the next page.)



Properties of the Task Module

**To Define a RuleRunner Task Module (continued)**

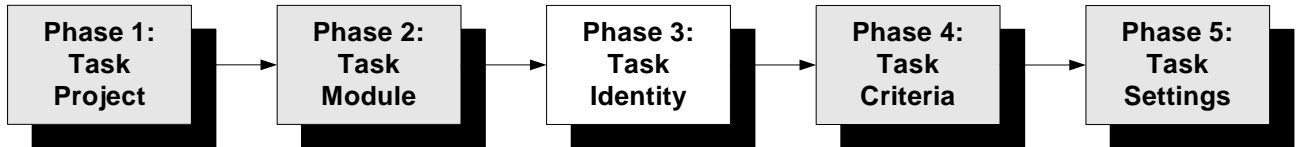
- | Step | Action  |
|------|---|
| 10.  | Press the Test button. If the connection between the Task Module and Task Project is secure, you will receive this technical message: |



- ✓ The Taskmaster Application Wizard provides a new application with a fully-defined Task Module for every task in each job, including *RuleRunner* tasks such as vScan, RuleRunner, Verify and Export. Be sure to take a moment to explore each module and the connection between its task and Task Project.

## RuleRunner Task Setup – Phase 3: Task Identity

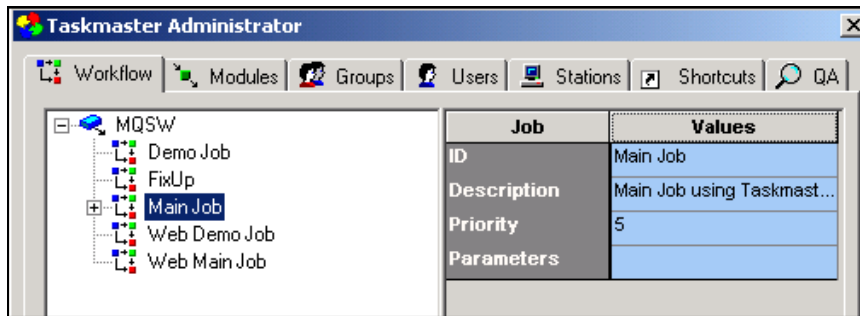
Phase 3 assembles the *RuleRunner* task as a *Taskmaster* component with a *clear identity* that reflects its purpose - and assigns the Task Module (Phase 2) that will connect the Task Definition to its Task Project (Phase 1).



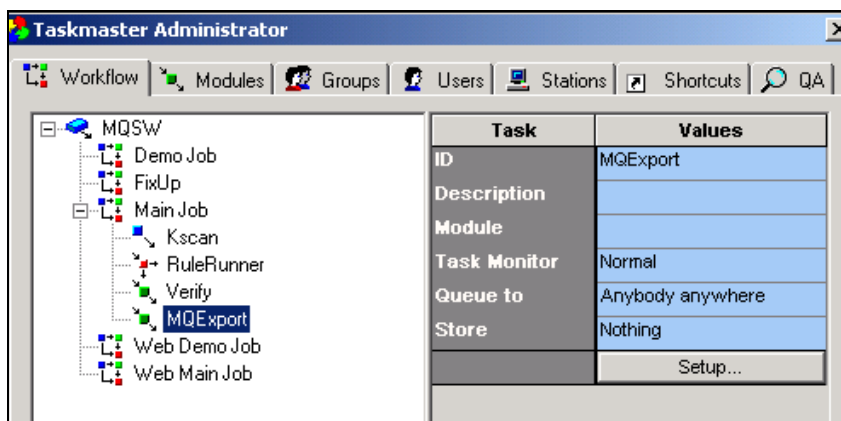
To provide the *RuleRunner* task with its identity:

Step	Action
------	--------

1. Open the *Taskmaster Administrator's* Workflow tab.
2. Right-click on the **job** which will contain the task (*Main Job*, in this example.)



3. Select **New** and **Task** from the options.
4. Enter a *unique* Task ID that indicates this *RuleRunner* task's role, in the open space below the Job ID. Be sure that this value appears in the **ID** field of the **Values** area as well.



---

**To Provide a RuleRunner Task with its Identity (continued)**


---

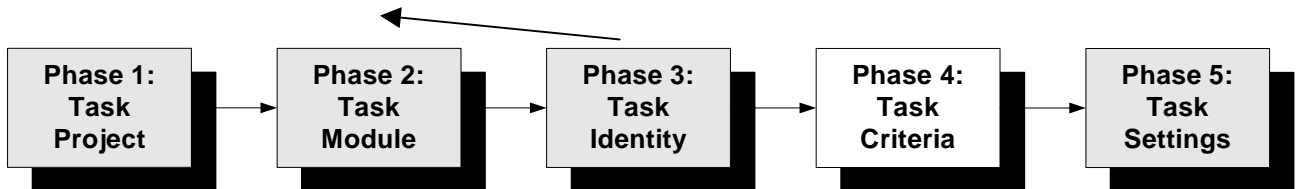
Step	Action
5.	In the <b>Values</b> area, enter a brief but important <b>Description</b> of this Verify task.
6.	From the <b>Module</b> drop-down list, select the ID of the Task Module you defined in Phase 2 (Page 11).
7.	Do not modify the default value of the <b>Task Monitor</b> property.
8.	Select a value other than <i>Anybody Anywhere</i> from the <b>Queue to</b> drop-down list <i>if</i> access to the new <i>RuleRunner</i> task is limited to a station and/or operator involved with an earlier task.
9.	Select a value other than <i>Nothing</i> from the <b>Store</b> field if tasks after this <i>RuleRunner</i> task can be limited to the station and/or operator running the current task.
11.	Press the <i>Taskmaster Administrator's</i> Apply button to save the identifying properties of the Task Definition.

- ✓ **Alert!** You'll find complete explanations of the Task Identity procedures in Chapter 6, or by pressing your F1 key when you are in the *Workflow* tab. This opens the set of *Taskmaster Help* topics that covers all aspects of the tab...including the elements of a Task Definition and of its **Queue to** and **Store** properties.
- ✓ Don't forget, too, that the Taskmaster Application Wizard provides an application with the skeletons of *RuleRunner* tasks of various types. For more about the wizard, see Chapter 2 and the *Application Wizard Guide*.

### RuleRunner Task Setup – Phase 4: Task Criteria

A *RuleRunner* task of any sort - even a vScan or Verify task - applies rules that you define in the *Rules* panel of your application's *Rule Manager Window* - well before you begin to setup the task.

Phase 4 determines exactly which rules will be applied by the task you're working with now - and the order in which they will be applied - as the task processes the current batch, its documents and pages, and the fields on each *source* page in the batch.



**Very important!** Before you begin this phase, be sure you have the following information:

- The Connection String that links the task to your application's Rules database.
- The names of the RuleSet Types with rules that the task is to apply.
- The order in which the task is to apply the rules of individual RuleSet Types.
- The names and paths of the Actions files (.rra) containing actions that make up the rules you've prepared.

To assign pre-defined rules to the *RuleRunner* task you're working with, highlight the task's ID in the *Taskmaster Administrator's Workflow* tab - and press the Setup button.

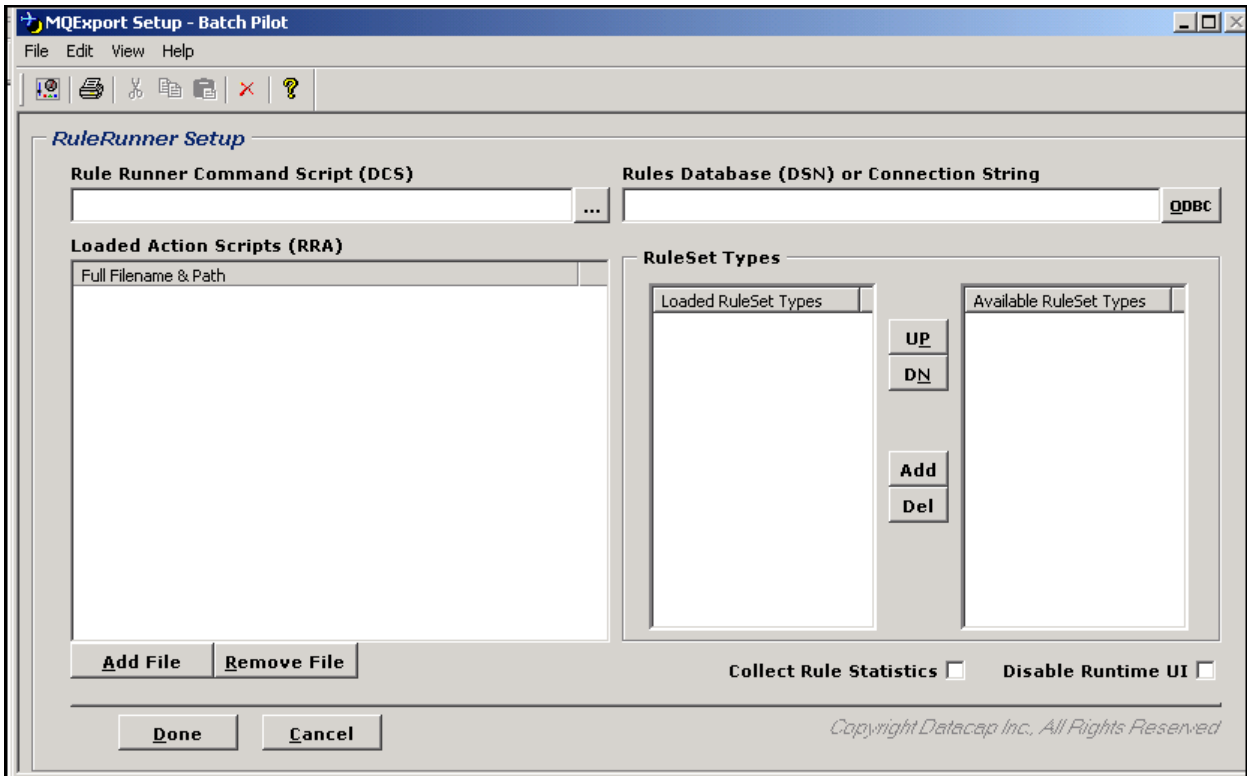
A completely empty *RuleRunner Task Setup* dialog may appear on your screen (see the illustration on the next page.) This is because there is no value in the **Rules Database** field.

- ✓ To correct the problem, enter the Connection String of your Rules database, or use the ODBC button to access the *ODBC Data Source Administrator* and select the applicable Data Source Name.

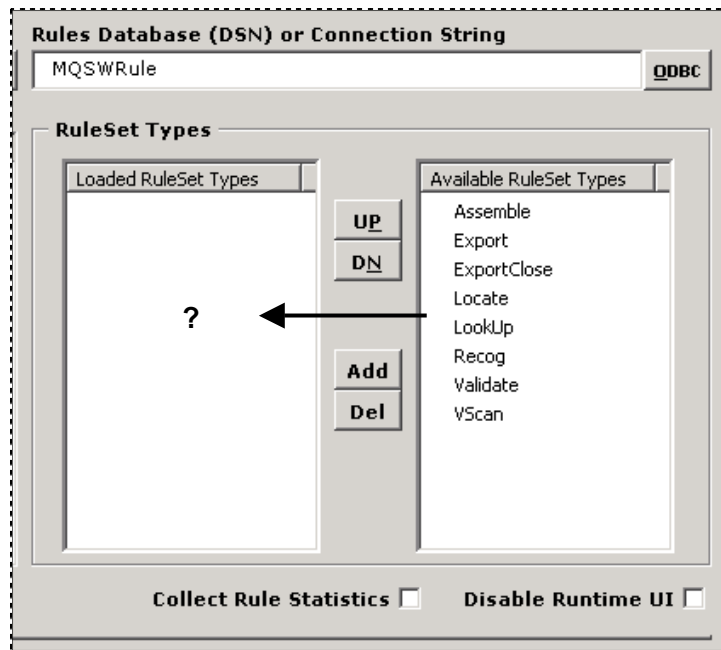
In the second example on the next page, entering the "MQSWRule" DSN has placed a list of the application's active and engaged RuleSet Types in the **Available RuleSet Types** field: rules of these types have been previously defined, and applied to objects of the Document Hierarchy.

The challenge at this point is to decide which types contain rules your new task will apply. Once you have made this decision, you can move RuleSet Type ID's from the **Available RuleSet Types** field to the **Loaded RuleSet Types** field.





RuleRunner Setup – MQExport task

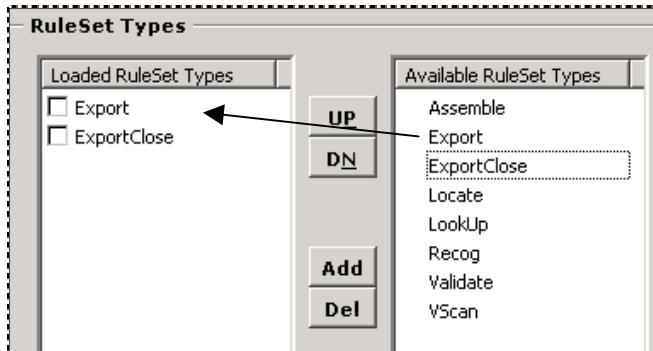


Available RuleSet Types – MQSW Application

## Setting up a RuleRunner Task

The limited scope of the MQExport task will result in a minimal selection involving these simple steps:

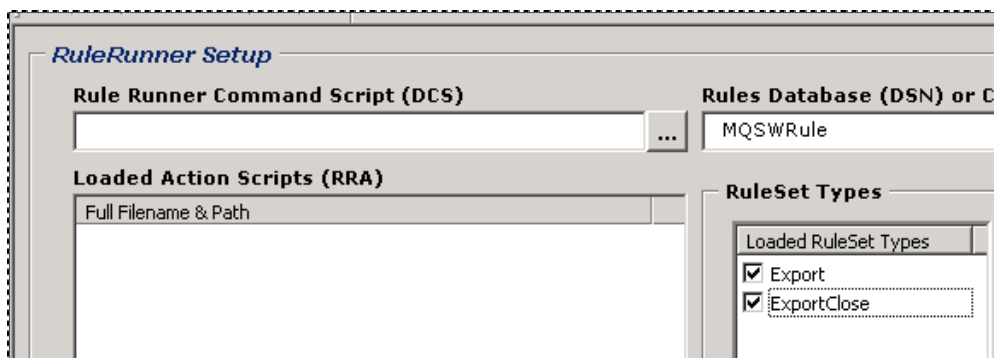
- ◆ Highlight the **Export** RuleSet Type in the **Available RuleSet Types** list.
- ◆ Press the Add button to move the RuleSet ID to the Loaded **RuleSet Types** list.
- ◆ Repeat this procedure for the **ExportClose** RuleSet Type.



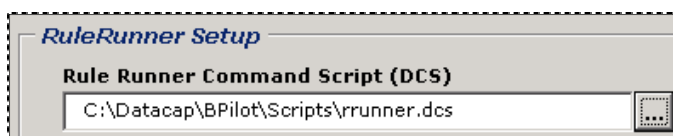
- ✓ In this example, the MQExport task *cannot* apply **Export** or **ExportClose** rules until you take these additional steps:

Step	Action
------	--------

1. Check the checkbox next to each RuleSet ID in the **Loaded RuleSet Types** field.
2. If necessary, use the UP and DN buttons to revise the order in which the MQExport task will apply rules of each type (rules at the top of the list are applied first!)

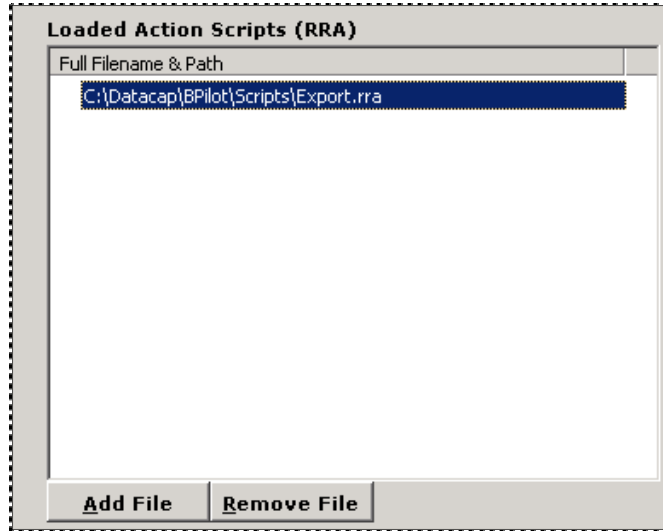


3. Use the Ellipsis button to provide the name and path of the **RuleRunner Command Script** file (.dcs). You can usually find this file in the **Scripts** subfolder of the **Datacap** directory's **BPilot** folder.

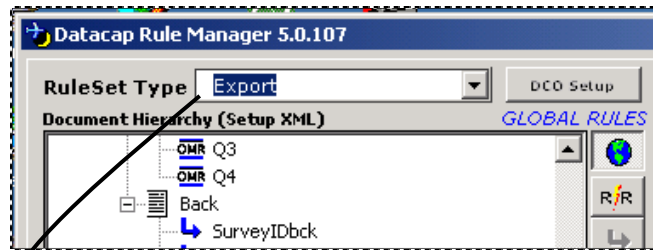


To Setup a RuleRunner Task (continued)

- | Step | Action   |
|------|--|
| 4.   | Use the Add File button to locate and add the names of the Actions file (.rra) that contribute to rules of the types you listed in the Loaded RuleSet Types field (Step #1). |



✓ **Warning!** Identifying *all* relevant Actions files may take some detective work. – and the place to start is in the *Rules* panel of your application’s **Rule Manager Window**.



Export RuleSet Type

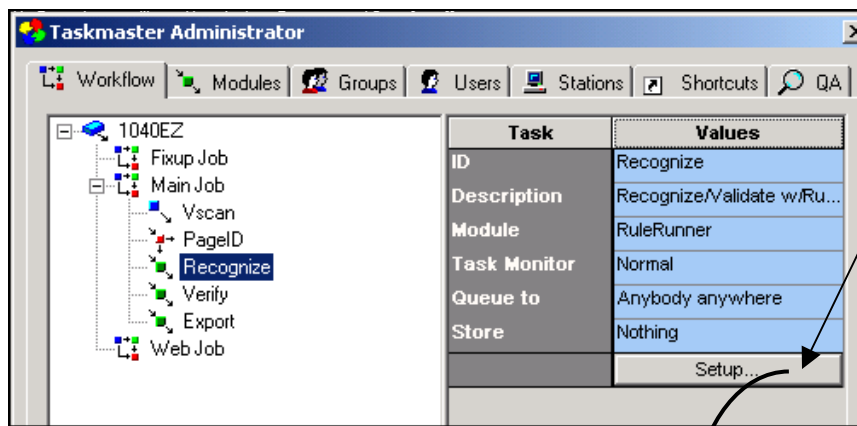
Export Actions Library	Source RRA
AbortOnError	rrunner
BatchVariable_ExportValue	Export
BlankFields	Export
BlankLines	Export
BPilot	Export
CheckAllIntegrity	rrunner
CheckAnchor	rrunner
CheckDocCount	rrunner

Export RuleSet Type Actions Library

### RuleRunner Task Setup – Phase 5: Task Settings

The closing phase of the *RuleRunner* task's configuration assigns values to specifications in the tabs of its *Task Settings* dialog.

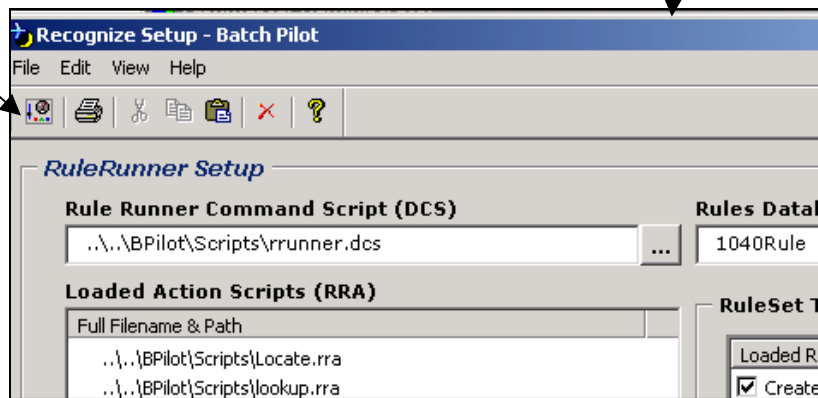
- ✓ To access this dialog:
  - Highlight the applicable Task ID in the *Taskmaster Administrator's Workflow* tab.
  - Click on the Setup button to open the *Batch Pilot Window* – and the *RuleRunner Task Setup* dialog.
  - Click on the **Task Settings icon** in the window's toolbar or select Task Settings from the *Batch Pilot Window's File* menu.



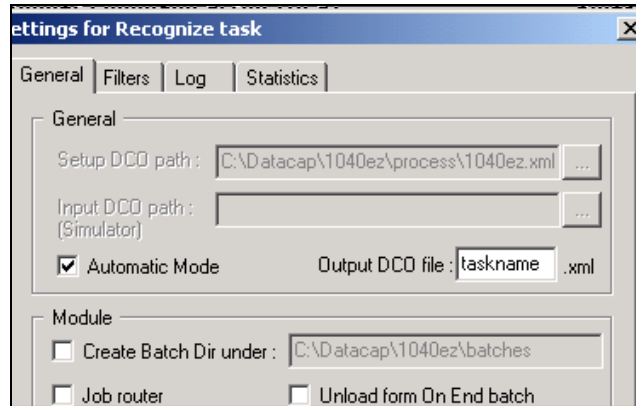
Press this button.

1040EZ Taskmaster Administrator

Click here.



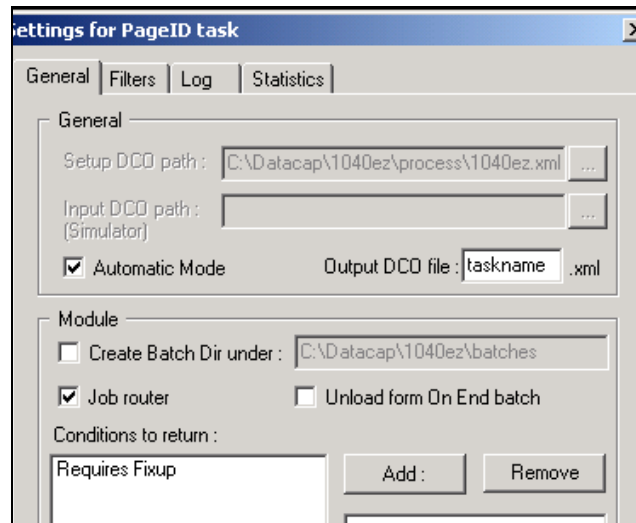
RuleRunner Setup dialog – *Recognize* task



**Task Settings dialog – General tab**

The values you assign to settings of the *General* tab (above) depend on the nature of the task itself.

- Because most *RuleRunner* tasks operate in the processing background without operator intervention, the **Automatic Mode** option is usually checked.
- If a task such as the *1040EZ* PageID task diverts problem batches to a FixUp job, the **Job Router** option is checked (Chapter 10):



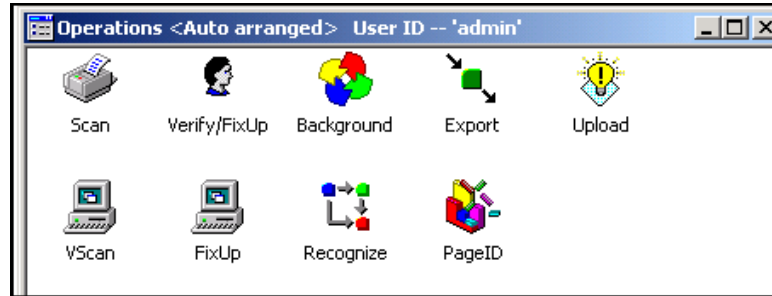
**Task Settings dialog – General tab  
PageID Task**

Chapter 6 describes the settings in each tab of the *Task Settings* dialog.

## RuleRunner Tasks – Operations

A typical *RuleRunner* task runs in the processing background when an authorized “user” – an operator, Supervisor or Administrator – clicks on the task’s icon in *Taskmaster’s Operations* window.

The *Operations* window of the *1040EZ* application provides separate Job-Task shortcut icons for three *RuleRunner* tasks: **PageID**, **Recognize** and **Export**.



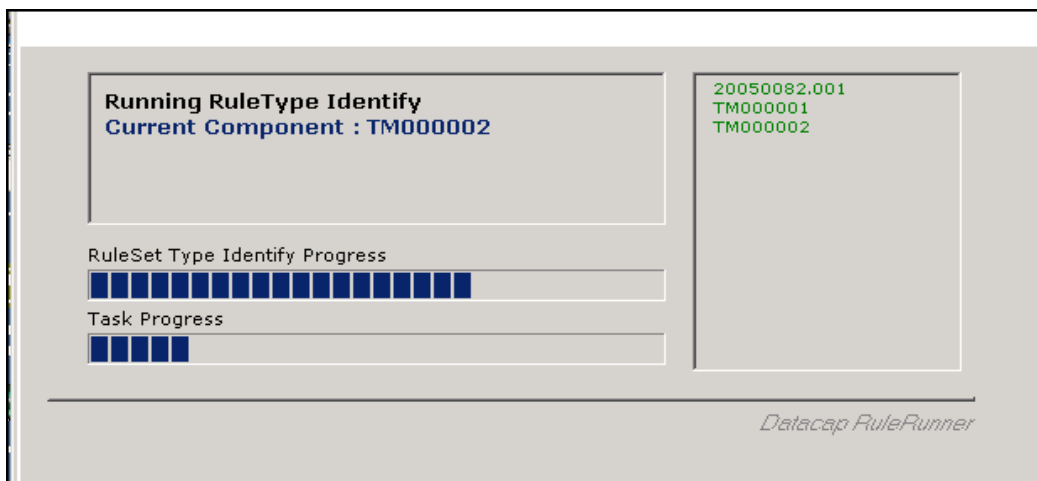
1040EZ Operations Window

*If...*

- ◆ You double-click on one of these icons – **Export**, for example; *and*
- ◆ There are batches in the task’s processing queue; *and*
- ◆ You have selected the **Automatic** option in the *General* tab of the Task Settings dialog (Page 20);

*Then* the Export task(s) covered by the icon will process *all* batches in their queues.

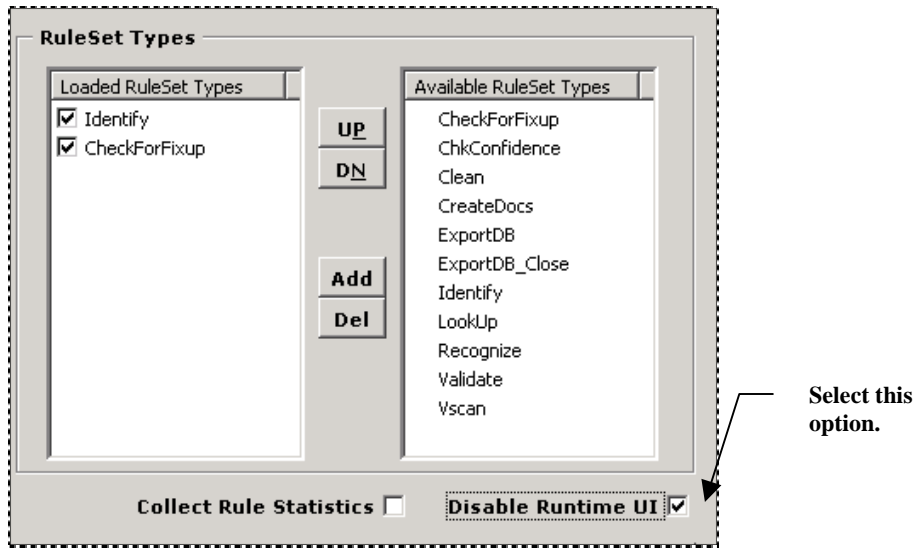
During processing, most *RuleRunner* tasks will display an *Operations* screen similar to this:



PageID Task – Operations screen

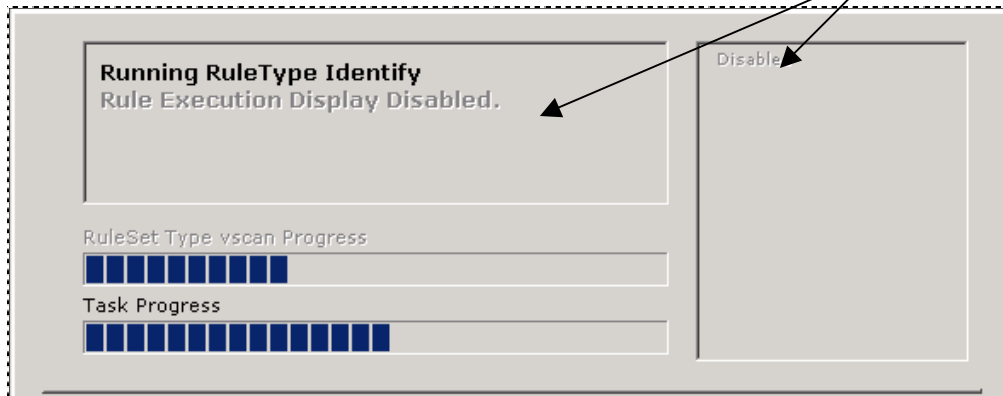
The *Operations* screen identifies:

- The RuleSet Type that the task is currently applying (**Identify**, in the example).
  - The component of the batch that is the object of the current rule’s attention (*TM000002* specifies a particular page.)
  - The progress of rules that belong to the **Identify** RuleSet Type as they process the page.
  - The progress of the task as it processes the batch.
  - Components of the batch. In this example, Batch *20060082.001* has just two pages.
- ✓ **Important!** You can “turn off” displays of RuleSet Types and Batch Components by selecting the **Disable Runtime UI** option in the task’s *RuleRunner Setup* dialog (illustrated below).



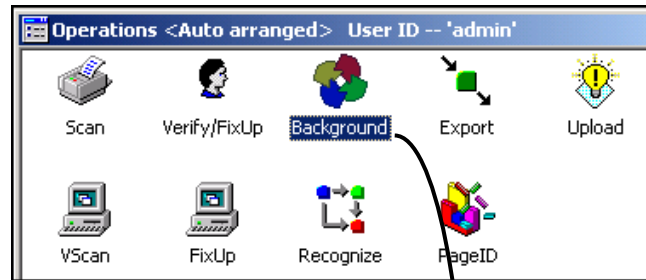
PageID RuleRunner Setup dialog

As a result, the screen will look like this: **Disabled!**

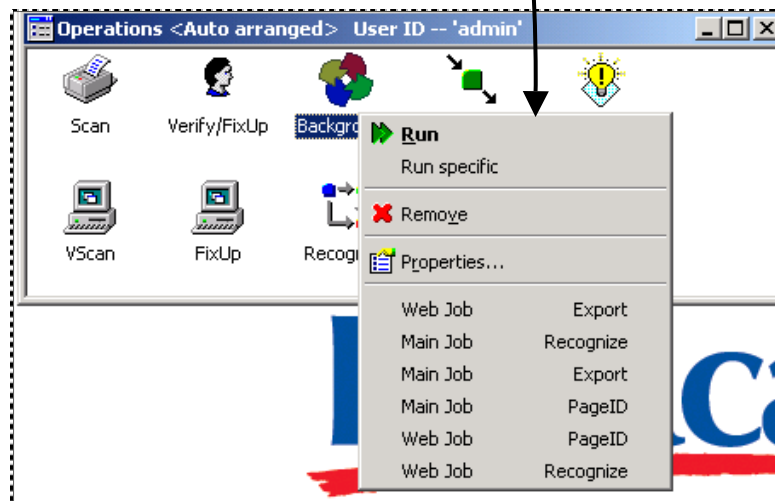


## RuleRunner Tasks – Operations

Double-clicking on the *1040EZ* application's **Background** shortcut icon automatically launches batches in the queues of the Job/Task Combinations illustrated below.



**1040EZ Operations Window**



**Properties of the Background Job-Task Shortcut**

Chapter 5 of the *Taskmaster Administrator's Guide* shows you how to define your application's Job-Task shortcut icons. Comparable explanations are instantly available from **Taskmaster Help** when you access the *Shortcuts* tab of your *Taskmaster Administrator* – and press the F1 button on your keyboard.