6

Creating and Modifying Rules

This chapter explains how the tabs and tools of Datacap Studio's **Rulemanager Page:**

- Modify the Ruleset/Rule combinations that determine how a *rulerunner* task processes a batch and its contents;

- Create new Rulesets and rules and bind the resulting Ruleset/Rule combinations to objects of the Document Hierarchy.

- Assign Rulesets to the tasks that will carry them out during the operations of a *Taskmaster RRS* application.
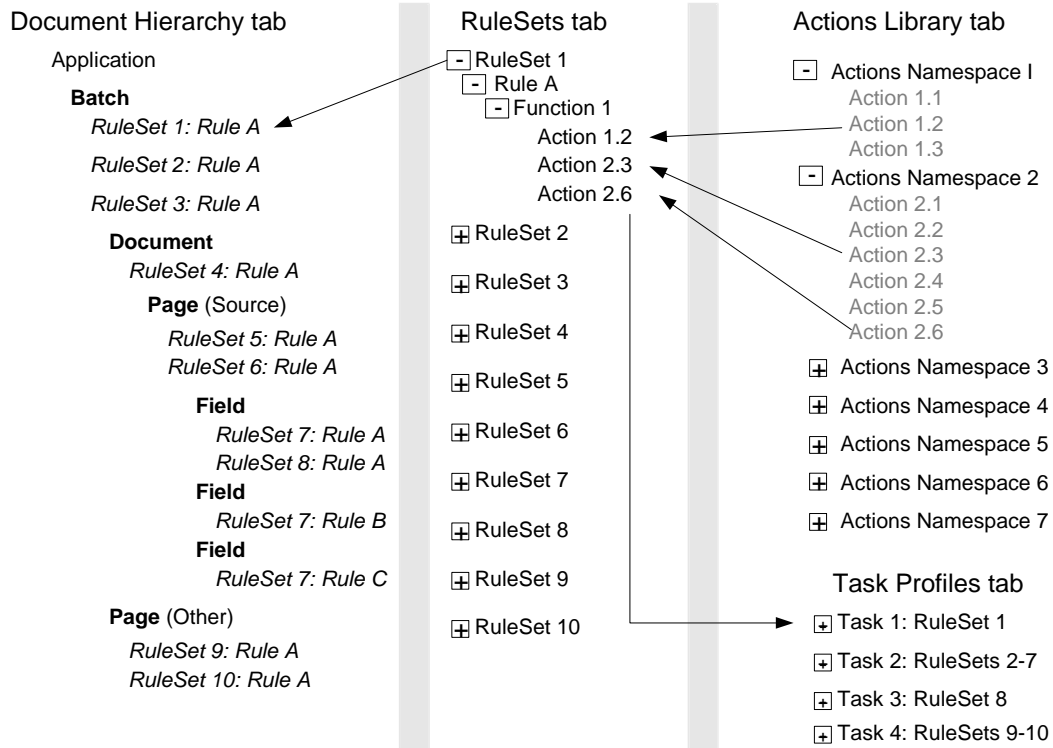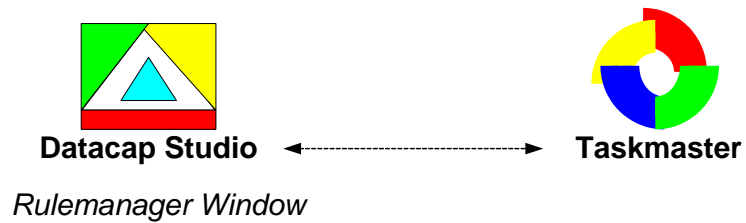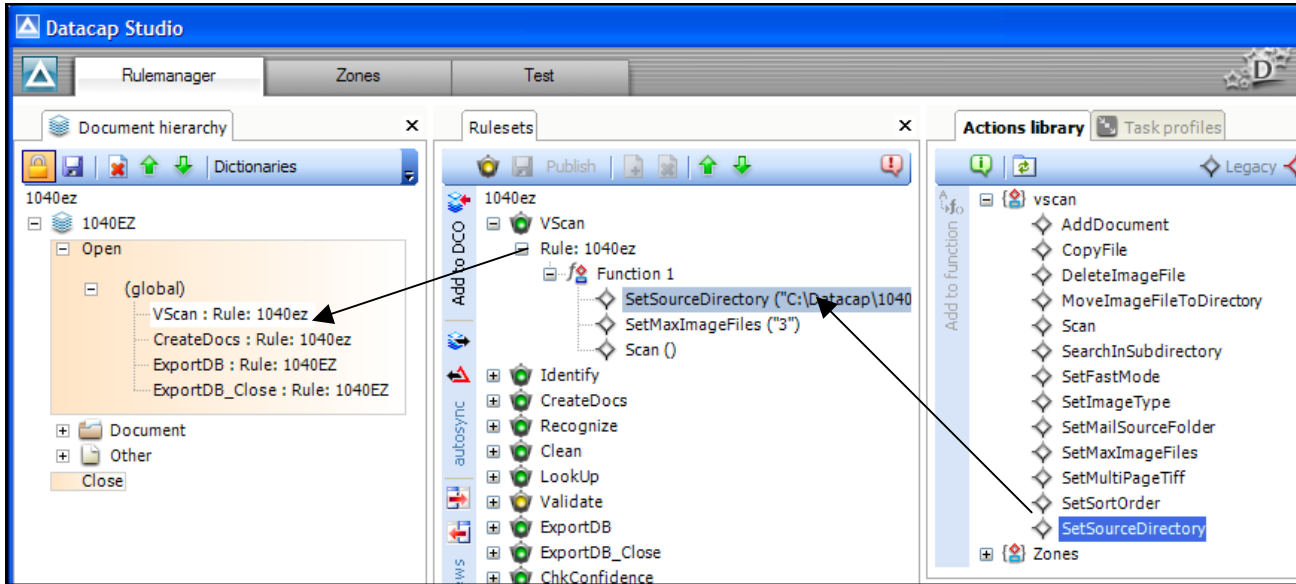
Chapter 6 covers these topics:

6.1 Rulesets and the Rulemanager Page

You use tools in the tabs of Datacap Studio's **Rulemanager Page** to:

- Assemble and modify your application's **Document Hierarchy** – also known as its *setup* DCO (Chapter 5).

- Define the **Rulesets** that will guide individual *Rulerunner* tasks.

- Define each **rule** that is part of a **Ruleset** – by assigning one or more **actions** to a function, and one or more **functions** to the **rule**.

- Bind a completed **Ruleset/Rule** combination to an object of the Document Hierarchy.

- Assign the Rulesets that contain these rules to *Taskmaster RRS* **Task Profiles**.

Datacap Studio ←--------------------------→ **Taskmaster**

Rulemanager Window

Document Hierarchy tab	RuleSets tab	Actions Library tab

Document Hierarchy tab

Application

Batch
RuleSet 1: Rule A
RuleSet 2: Rule A
RuleSet 3: Rule A

Document
RuleSet 4: Rule A
Page (Source)
RuleSet 5: Rule A
RuleSet 6: Rule A

Field
RuleSet 7: Rule A
RuleSet 8: Rule A
Field
RuleSet 7: Rule B
Field
RuleSet 7: Rule C

Page (Other)
RuleSet 9: Rule A
RuleSet 10: Rule A

RuleSets tab

[-] RuleSet 1
  [-] Rule A
    [-] Function 1
      Action 1.2
      Action 2.3
      Action 2.6

[+] RuleSet 2
[+] RuleSet 3
[+] RuleSet 4
[+] RuleSet 5
[+] RuleSet 6
[+] RuleSet 7
[+] RuleSet 8
[+] RuleSet 9
[+] RuleSet 10

Actions Library tab

[-] Actions Namespace I
  Action 1.1
  Action 1.2
  Action 1.3
[-] Actions Namespace 2
  Action 2.1
  Action 2.2
  Action 2.3
  Action 2.4
  Action 2.5
  Action 2.6
[+] Actions Namespace 3
[+] Actions Namespace 4
[+] Actions Namespace 5
[+] Actions Namespace 6
[+] Actions Namespace 7

Task Profiles tab

[+] Task 1: RuleSet 1
[+] Task 2: RuleSets 2-7
[+] Task 3: RuleSet 8
[+] Task 4: RuleSets 9-10

1040EZ DStudio: Rulemanager Page

Above, on the left, the *Document Hierarchy* tab displays data of two kinds:

♦ Identities of objects at each level of the Document Hierarchy: **Batch**, **Document**, **Page** and **Field** (Chapter 5).

♦ Ruleset/Rule combination(s) assigned to an object of the Document Hierarchy – in this case, to the *1040EZ* **Batch** object.

The *Rulesets* tab in the middle of the example lists:

♦ Rulesets of the application

♦ Rules that belong to a Ruleset: **Rule:1040ez**, for example, is a member of the **VScan** Ruleset.

♦ Functions within a rule: **Function1** of **Rule:1040ez**,

♦ Action(s) that are the function's code:

```
SetSourceDirectory("C:\Datacap\1040EZ\images")
SetMaximumFules("3")
Scan()
```

Tools of the *Rulesets* tab construct Rulesets and rules, and link them to objects of the Document Hierarchy.

The *Actions Library* tab on the right lists available Actions categories, and the actions that belong to each category. Tools of this tab assign actions to the function(s) of a rule.

The *Task Profiles* tab, which is behind the *Actions Library* tab, lists Rulesets that will be applied by a specific *rulerunnner* task when the task runs. You use this tab to assign Rulesets to a task after the task's has set up by the ***Taskmaster Administrator*** (Page 22).

✓   **Please note:** Effective management of the **Rulemanager Page** is a process that involves these steps:

1. Use the *Document Hierarchy* tab to set and modify your application's Document Hierarchy (Chapter 5).

2. Be sure that your Actions Library is complete…that the *Actions Library* tab lists all the actions you need (Page 5).

3. Use the *Rulesets* tab to define Rulesets, and rules that contain actions selected from the Actions Library (Page 12).

4. Work with both the *Rulesets* tab and the *Document Hierarchy* tab to "bind" Ruleset/Rule combinations to objects of the Document Hierarchy.

5. With the *Task Profiles* tab, establish or modify the Task Profiles that assign specific Rulesets to specific tasks of a particular job (Page 22).

The presentation of the explanations in Chapter 6 mirrors this sequence.

6.2 Rulemanager Page: Actions Libraries

Within Datacap Studio's **Rulemanager Page**:

♦ A **Ruleset** consists of **rules**. Rules consist of **functions** – and every function must have one or more **actions**.

♦ Actions are organized and documented in the *Actions Library* tab. Tools of this tab also assign actions to the functions of individual rules.

♦ A rule is part of a **Ruleset**. Both are defined in the *Rulesets* tab. Together, they form a **Ruleset/Rule combination**.

♦ Tools of the *Rulesets* tab "bind" Ruleset/Rule combinations to individual objects of the Document Hierarchy.

✓ The direction of this flow means that your work with the rules components of Datacap Studio begins with actions – in a sequence that locates Actions files; reviews the contents of these files, and checks the accompanying Actions documentation.
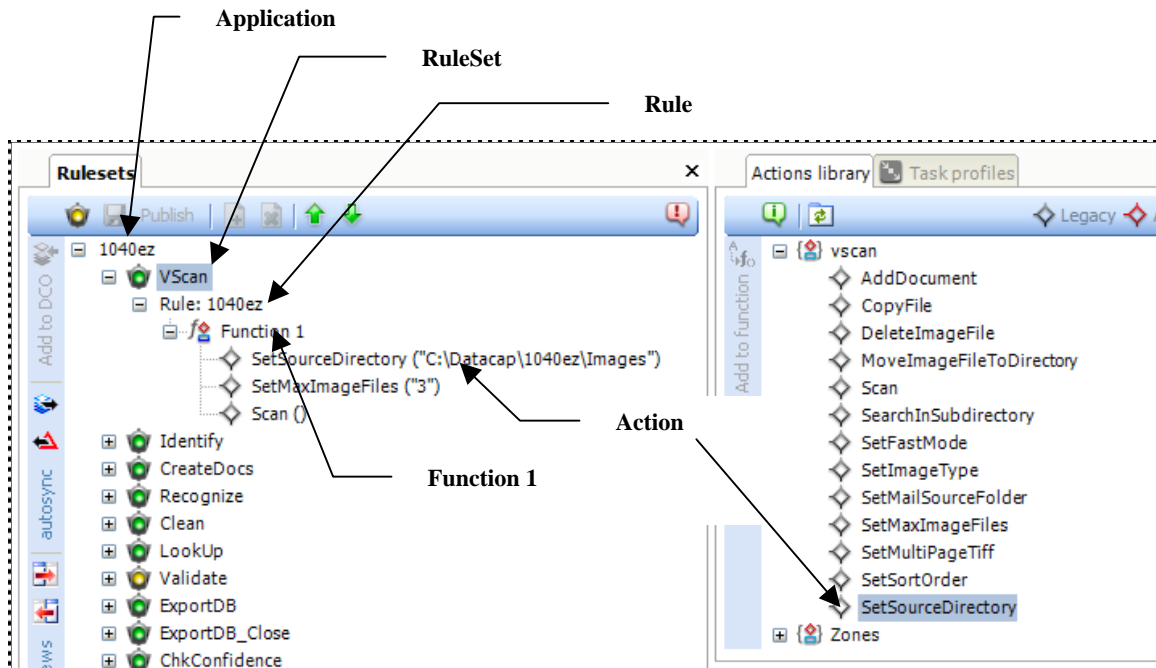

1040EZ VScan/Rule: 1040ez – *Function1*
**Set Source Directory Action**

6.2.1 Actions

✓   In the *Actions Library* tab, if you right-click on an Actions Category, Datacap Studio
     provides a brief description of the category.
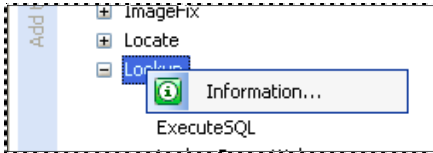
If you right-click on an action's title, DStudio describes the action and alerts you to any
parameter(s) it requires.

Action

LookupCurrrentValue Action

Parameter

Actions files (.rrx)

Export Actions Library
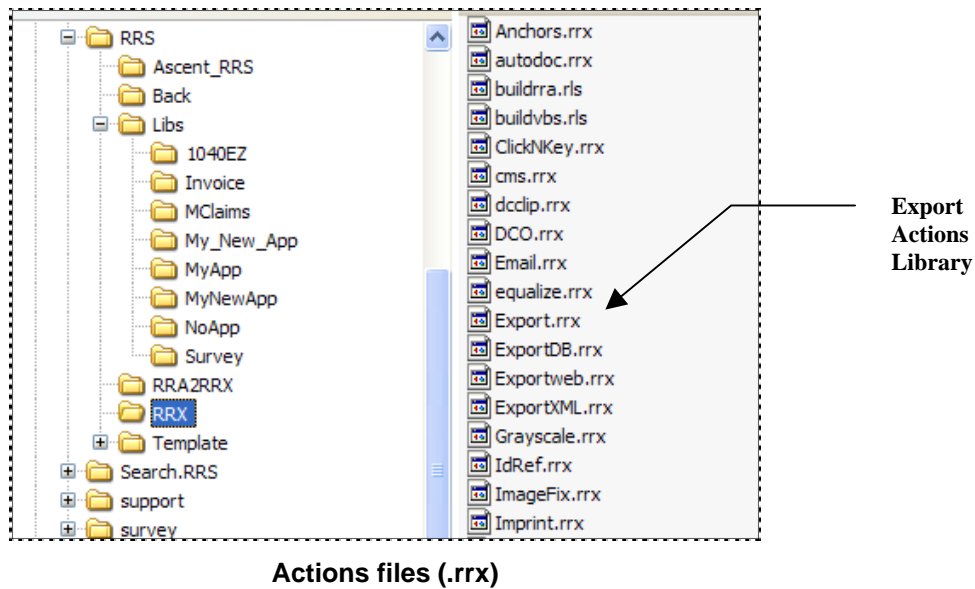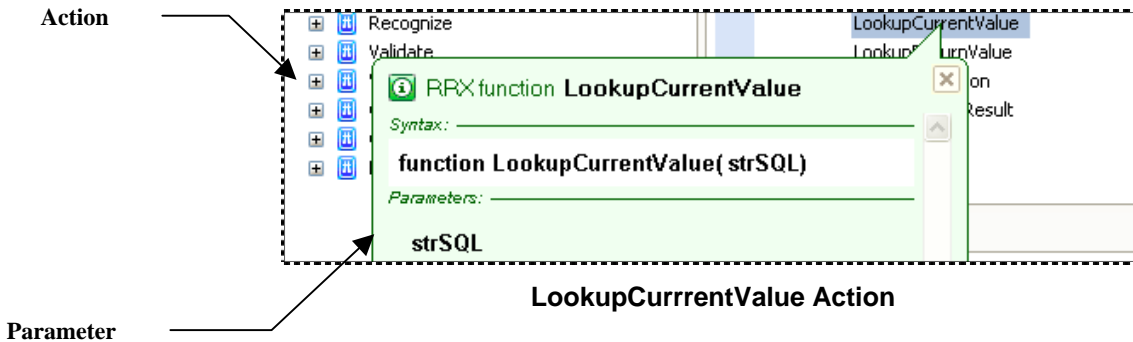
Actions are stored in Actions Library files (.rrx), in the **RRX** sub-folder of your **Datacap**
directory's **RRS** folder (illustrated on the next page.)
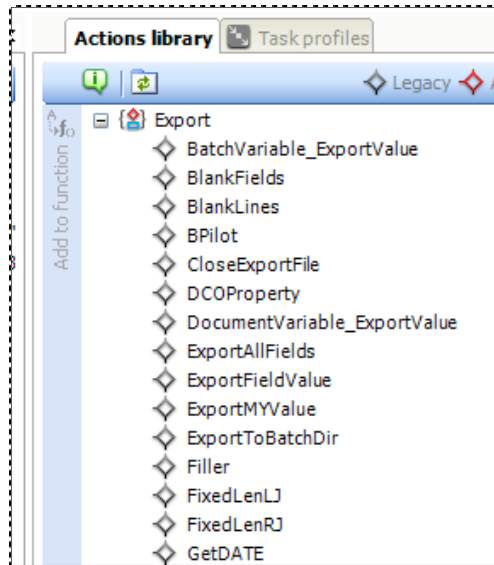
Export Actions (partial list)

6.2.2 Toolbar Icons of the Actions Library Tab

This tab has only just two active icons:

- The icon at the top, on the left, retrieves information about the procedure you have highlighted.

- The **Add to Function** icon on the left-hand edge assigns the procedure you have highlighted to a function in the *Rulesets* tab (Page 10).
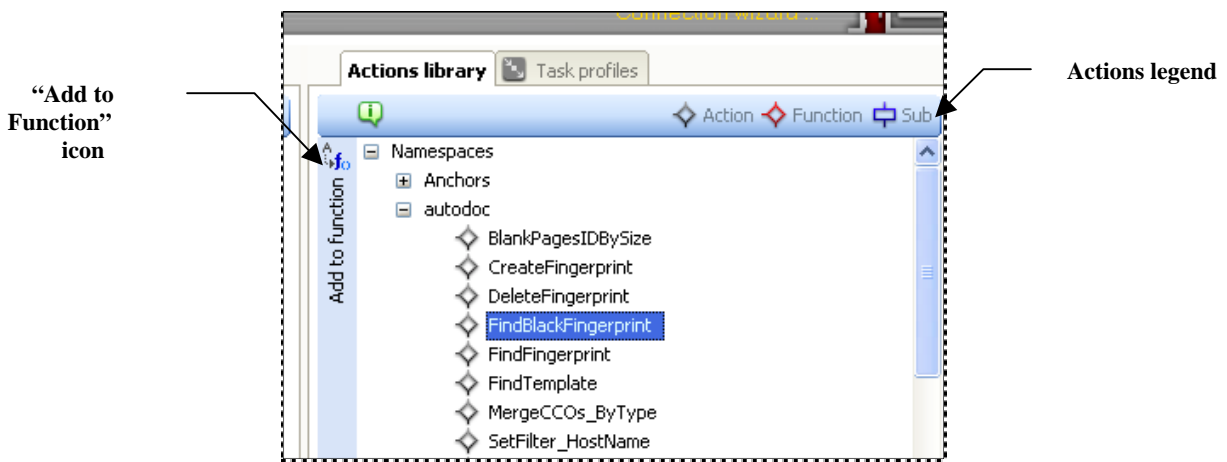
Actions Library tab

In addition, the Legend on the right, identifies the kind of action you have selected:

♦ An **action** is a VBScript procedure that returns a Boolean (*True* or *False*) value. (These are not standard actions.)

♦ A **function** is a VBScript procedure that returns a *Boolean* value, or a *String* value.

♦ A **subroutine** is a VBScript procedure that does not return a value.

6.2.3 How to Add an Actions Library to Datacap Studio

Datacap Taskmaster Installation automatically supplies a full set of up-to-date Actions Libraries (.rrx). *Alert!* Although you can add new libraries, and new actions to a library – be sure to consult Datacap Support or your Installation Manager first. (This section only explains the basic principles involved.)

Each RRX file contains related actions in a unique category.

✓ In the case of a new Actions Library file, for example, take these steps to add it to Datacap Studio:

• Give the file a name that reflects the nature of the actions it contains.

• Copy the file.

• Paste it into the **RRX** sub-folder of the **Datacap** directory's **RRS** folder.

• Open your Datacap Studio.

Be sure the new file is part of the *Actions library* tab'**s** list of files with *global* actions.

This very short example from the **Validations.rrx** file is the code for the single **IsThisFieldFilled** action.

```
<af name="IsThisFieldFilled" access="public" bInter="bInter"
bDebug="bDebug" qi="Confirms if current field has a captured
value.">
<h><![CDATA[ Confirms if the current field has a captured
value. ]]>
<e><![CDATA[ IsThisFieldFilled() ]]></e>
</h><![CDATA[
    If(CurrentObj.ObjectType = DCO_FIELD) Then
        If(Trim(CurrentObj.Text)<>"") Then
            IsThisFieldFilled = true
        Else
            IsThisFieldFilled = false
        end if
    Else
        writelog("function IsThisFieldFilled works only on
field level")
        IsThisFieldFilled = False
  End if ]]></af>
```

If you do add an action, Datacap Studio will list it as part of the Actions Library. **Remember!** Be sure to work with Datacap Support or your Implementation Manager if you intend to add a new action to an Actions Library.

✓ The *Properties* dialog does not display properties for a library or an action until the action is part of a rule (Page 12).
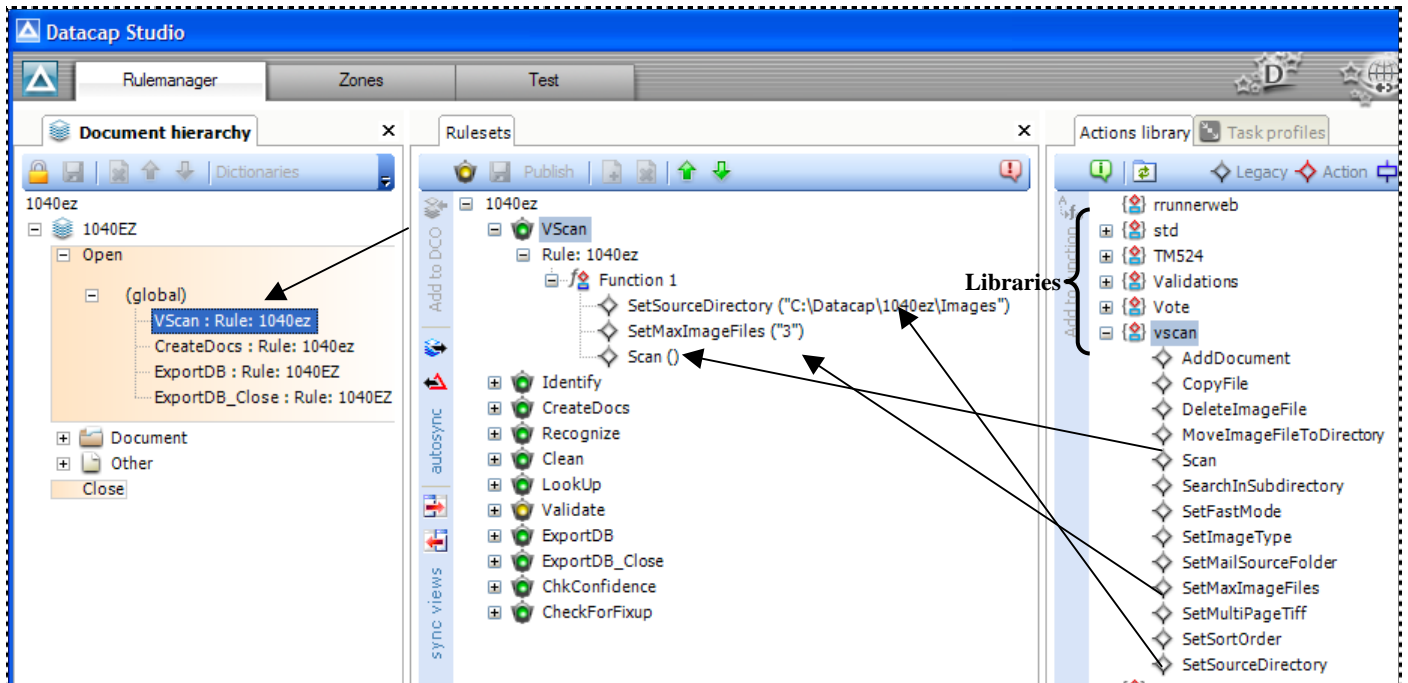
6.3 Rulemanager Page: Rulesets Tab

You group rules into logical groups in the *Rulesets* tab, then build logic into the rules themselves.

In the diagram below, the developer has used tools of the *Rulesets* tab to:

- Set up the *VScan* **Ruleset**.
- Add a **rule** – *Rule:104ez* – to the *VScan* **Ruleset**.
- Provide the rule with one **function**: *Function 1*.

The **Add to Function** icon of the *Actions Library* tab then assigns the action to the rule's function. Afterwards, the *Rulesets* tab binds the new Ruleset/Rule combination to the appropriate object of the Document Hierarchy.

1040EZ Rulemanager Page

✓ *Very important!* Pay close attention to the structure and wording of this sample Ruleset/Rule combination. You specify the titles of individual rules and functions. However, clarity and consistency in your naming conventions pays off when you are testing the Rulesets, rules and actions you have prepared (Chapter 8).

6.3.1 Icons of the Rulesets Tab

The table below describes icons in the toolbar along the top of the *Rulesets* tab:

Icons of the Rulesets Tab

| Icon	Description
	A Ruleset has a "public" version and a "private" version: the *Properties* dialog displays the current public and private Version ID's. For example, if you select the *1040EZ* application's **Clean** Ruleset before you click on the Lock icon, the *Properties* dialog will display two values: Locked Version <None> Public Version 0.0.0.1 After you click on the **Lock** icon, these values will be Locked Version 0.0.0.2 Public Version 0.0.0.1
	Saves definitions of new Rulesets and rules, and any changes you make.
	Makes a Ruleset/rule combination available for others to use.
	Adds a new Ruleset, rule or function. As examples, the **+** icon provides the Application ID with a new but untitled Ruleset; the Ruleset with a new but untitled rule; and a rule with a function. This icon does not *add* placeholders for a function's actions.
	Removes nodes at any level of a Ruleset – including the Ruleset itself. To reverse an inadvertent delete, unlock the Ruleset without saving it. *Alert!* This icon does not warn you first.
	The toolbar's arrows move actions *only* up and down within a function, or functions within a rule.

✓ The *Rulesets* tab has another, important icon at the top of its left edge: **Add to DCO**. The icon binds a Ruleset/Rule combination directly to an object of the Document Hierarchy (Page 12).
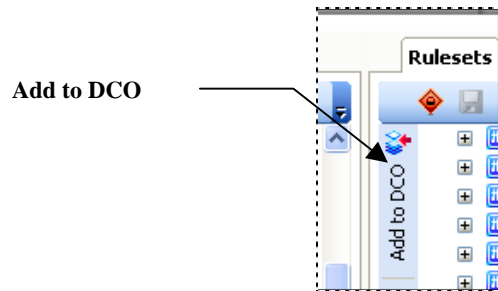
The other icons on the tab's left-hand edge are described on Page 20.

6.3.2 How to Define Ruleset⁄Rule Combinations

The *1040EZ* application's Datacap Studio lists Rulesets similar to those in this illustration.
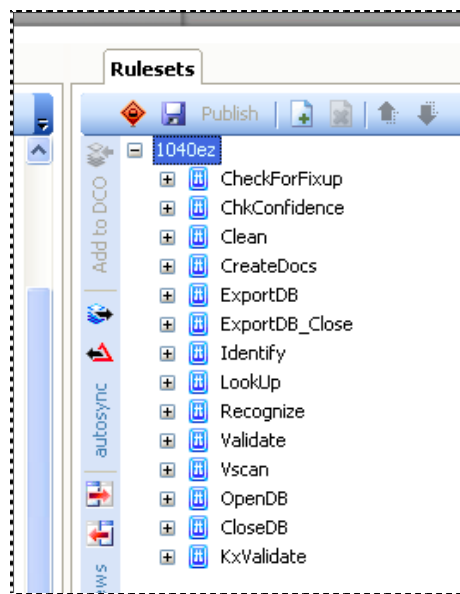
1040EZ Rulesets

When you press the + icon at a Ruleset's left edge, its rules line up. You can then open a rule to reveal its functions, and a function for details of its actions.

✓ The following pages show you how to set up a complete Ruleset hierarchy, beginning with a new Ruleset.

Rules of the 1040EZ Clean Ruleset
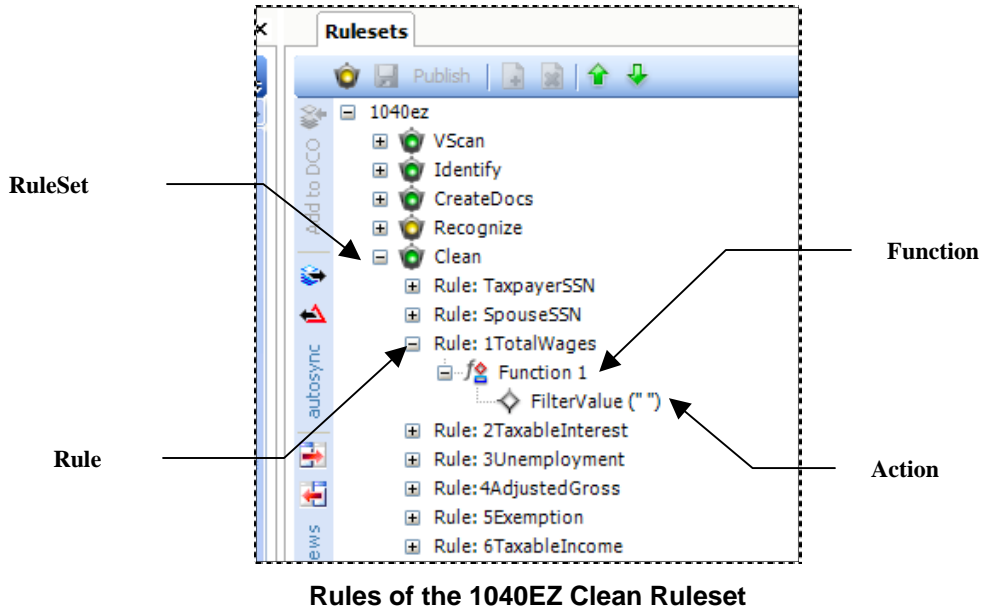
✔ A Ruleset/Rule combination is a hierarchy with this fundamental structure:

```
Ruleset <name>
    Rule: <object Name>
        Function 1
            Action(parameter)
            Action(parameter)
        Function 2
            Action(parameter)
```
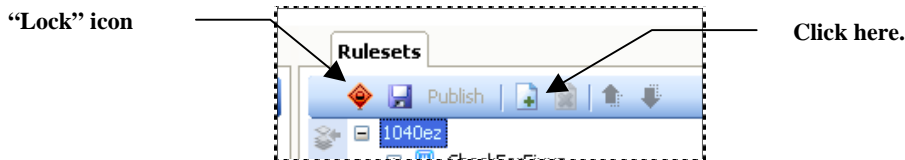
✔ In the following explanation, you can go right to Step #8 if you are working with an existing Ruleset.

To define a new Ruleset/Rule combination (using *1040EZ* and its Datacap Studio for practice), take these steps:

Step	Action
1.	Highlight the **Application's** title at the top of the *Rulesets* tab ("1040ez" in this example.)

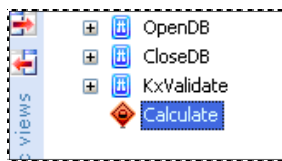| Step	Action
2.	Click on the Child (+) icon.

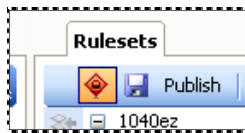### How to Define a New Ruleset/Rule Combination (continued)

Step	Action

3. Confirm that "New Ruleset" has been added to the bottom of the Rulesets list. (Notice that it is automatically "locked".
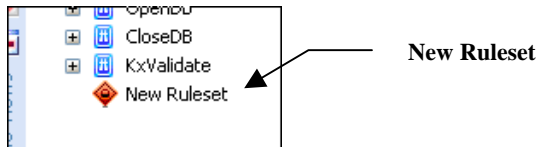
New Ruleset

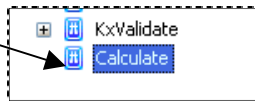4. Click once on the name, and modify it to meet your needs.

5. Note that the ◆ symbol remains. This is because the new Ruleset is locked.
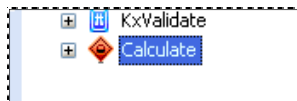
6. Click on the **Save** icon at the top.

7. Confirm that DStudio has changed the accompanying graphic.

Completed!

8. Press the **Child** (+) icon once again. This will add an empty rule to the new Rulesety – and place a "collapsed" icon next to the Ruleset node.

9. Open the Ruleset (*Calculate* in the example).

10. Assign a title to the new rule. (The convention used in the samples provided by Datacap combines "rule" with the target object's name. Keep in mind that you can prepare many rules for the new Ruleset **)**

How to Define a New Ruleset/Rule Combination (continued)

| Step	Action
11.	Press the **Child (+)** icon at the top of the tab to add a **function** to the rule.
12.	Review the function's title. (Step #12 automatically gives the function a title. Because a rule can have multiple functions, we advise you to keep these simple ID's.)
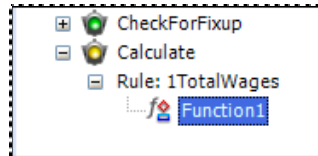
Step	Action
13.	Highlight the function's title and move to the *Actions Library* tab (Page 5).
14.	Select the first action that you intend to add to the function.
15.	Click on the tab's **Add to Function** icon.
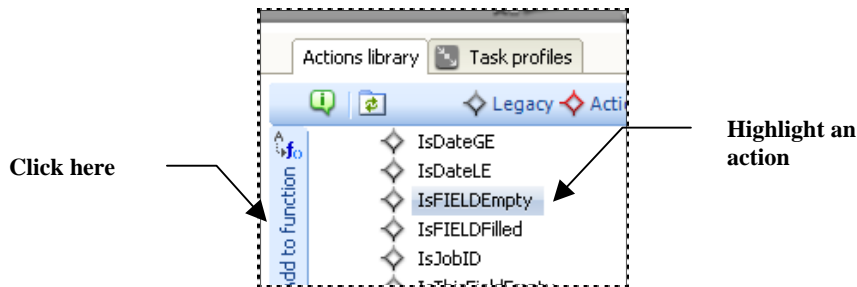
Step	Action
16.	Confirm that this action is now part of the function.
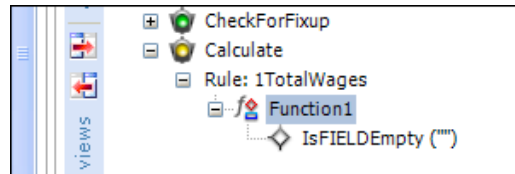
Step	Action
17.	Repeat Steps 16-18 for each additional action.
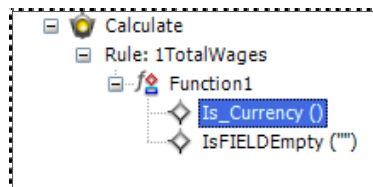
| Step	Action
18.	At the top of the *Rulesets* tab, press the **Save** icon.
19.	To give others access to the Ruleset/rule combination, click on the **Publish** icon and the **Lock** icon.

6.3.3 Binding Ruleset/Rule Combinations

A Ruleset/Rule combination has no value until you bind it to an object of the Document Hierarchy. (Chapter 5 describes all aspects of an application's Document Hierarchy.)

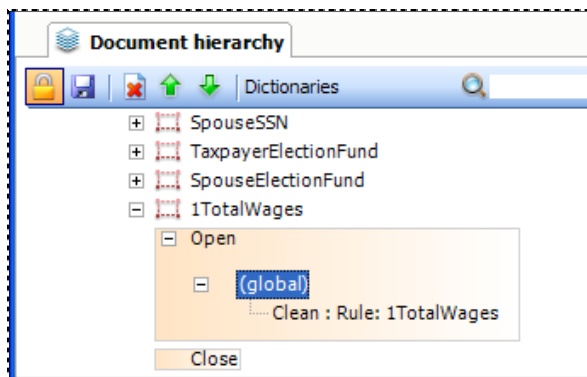Take the steps below to bind a Ruleset/Rule combination to an object of the Document Hierarchy:

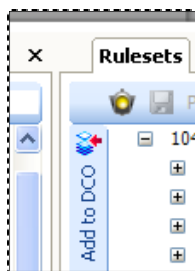| Step	Action
1.	Toggle the **Lock** icon of the *Document Hierarchy* tab to its *Lock* mode (Chapter 5).

Click here!
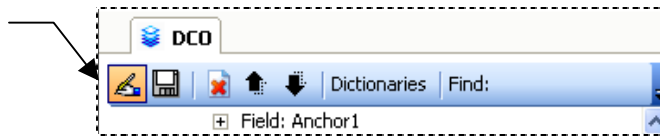
2.	Open the Document Hierarchy until you reach the object to which the Ruleset/Rule combination will be bound.
3.	Highlight the object's *(global)* attribute.

4.	In the *Rulesets* tab, select the new rule.
5.	Click on the **Add to DCO** icon (which is now available.)

6.	Confirm that Datacap Studio has assigned the Ruleset/Rule combination to the object you specified.
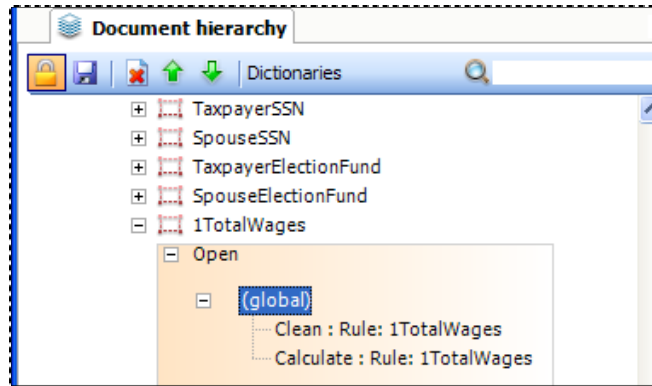
**New Ruleset/Rule Combination**
Calculate: Rule: 1TotalWages

✓ *Alert!* For the purposes of illustration, two "errors" are intentionally buried in the preceding example:

    1. In Step #21, the **IsFIELDEmpty** action should probably precede the **Is_Currency** action

    2. The **IsFIELDEmpty** action's syntax is `IsFIELDEmpty("")`. This means that the action requires a parameter.

The *Document Hierarchy* tab lists an object's Ruleset/Rule combinations but it does not show details of functions and actions. As Page 20 explains, you can go right to this information by highlighting the Ruleset/Rule combination in the *Document Hierarchy* tab and clicking on the **Sync Ruleset View with Document Hierarchy View** icon:

In this example, the problems remain.
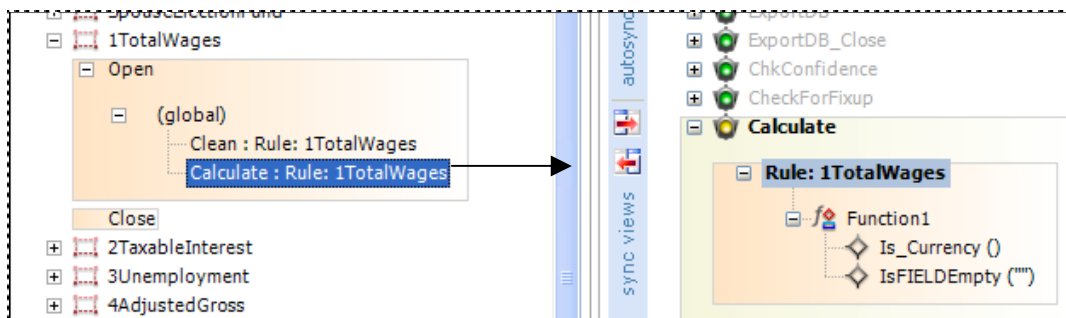
To move the second action up one position, highlight the **IsFIELDEmpty** action and click on the Up arrow.
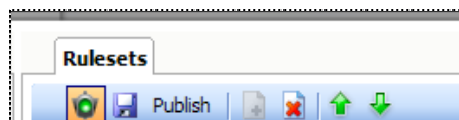
✓   Fixing the "missing parameter" problem is a little more challenging and is covered in the next section.

6.3.3.1 Action Parameters

In this example, a new *1040EZ* Ruleset/Rule combination uses **Locate** actions to find Zip Codes in the *Zip* fields of the *Page_1040ez* pages, and checks the fields' values. (Remember to experiment freely with the *1040EZ* application – you cannot destroy it!).
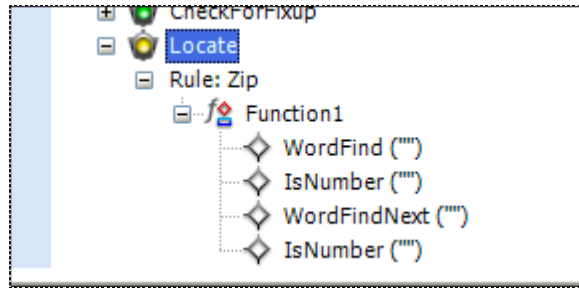

Locate: Rule; Zip

✓   The (" ") attached to the Action IDs in this list indicates that they are missing required parameters. If you right-click on the Action ID in the *Actions Library* tab, the action information will explain the nature of required parameters.
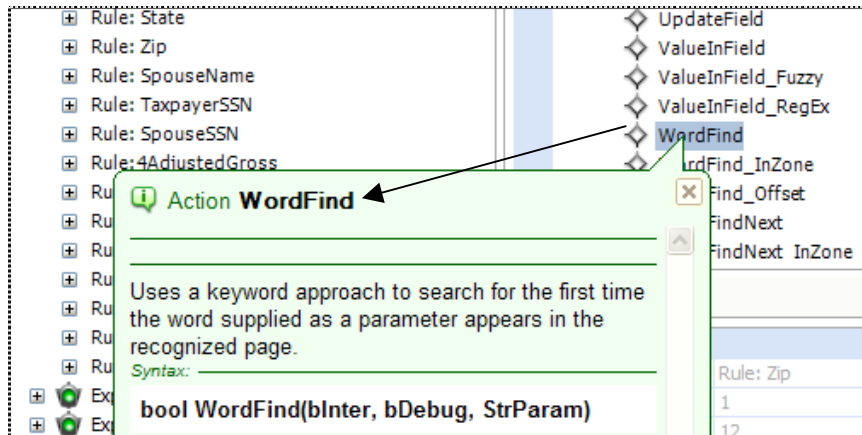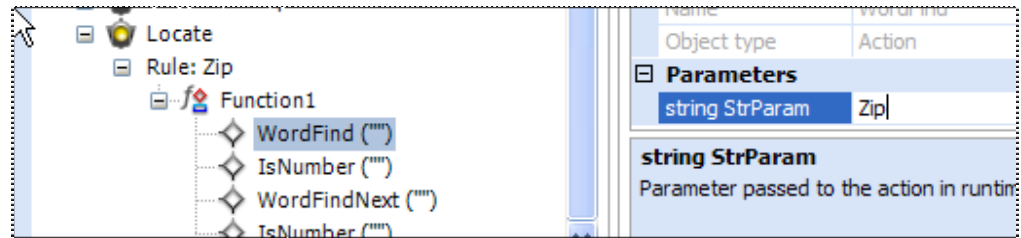

Action Information – WordFind Action
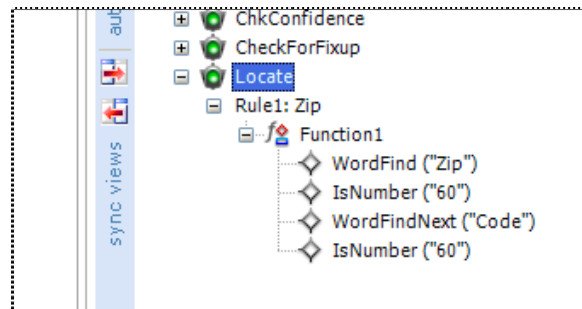
To add a parameter to an action, take these steps:

Step	Action
1.	Click on the *Ruleset* tab's **Lock** toggle. (**Remember:** Don't ignore this first step!)
2.	Right-click on the Action ID in the A*ctions Library* tab to review the action's parameter and syntax.
3.	Go to the function that includes the action.
4.	Highlight the action.
5.	Note that the ***Properties*** dialog does not have a parameter for the action. However, it does call for a *String* parameter.

String parameter
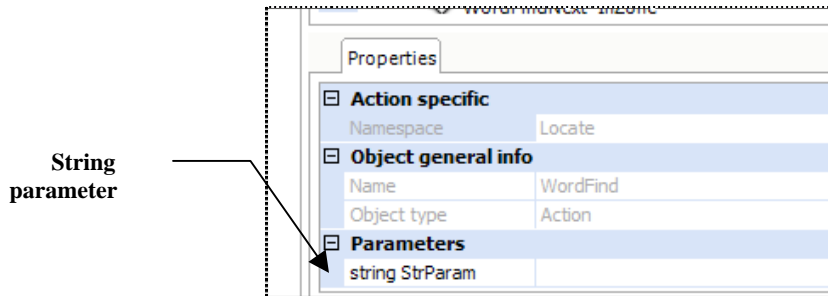
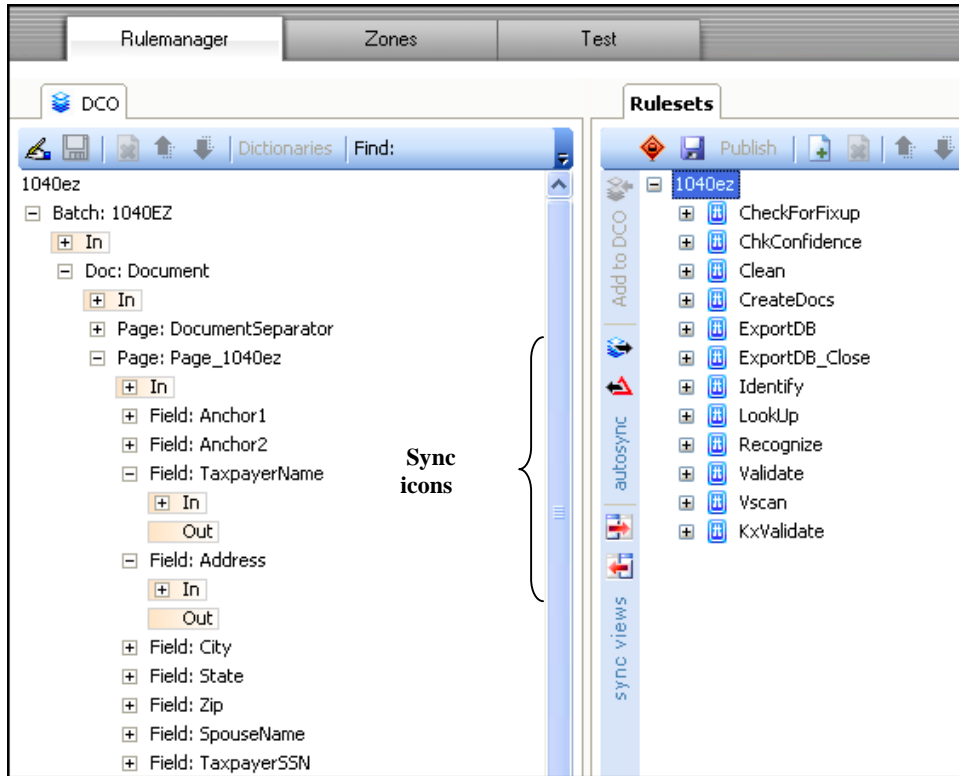| 6. | Place your cursor just to the right of the parameter's designation. |
| 7. | Enter the parameter and click on the Action ID in the function's listing. |

| 8. | Repeat these steps for other actions that need parameters. |

| 9. | Press the *Ruleset* tab's **Save** icon. |

6.4 Sync Icons of the Rulemanager Page

Here is a picture of the *1040EZ* application completely at rest – almost!

1040EZ Datacap Studio – *Rulemanager Page*

Certainly, the list of Rulesets in the *Rulesets* tab on the right is simple and cohesive.

On the left, the combination of +   and   –   indicators, along with the **Ins** and **Outs,** generate two very different types of information:

- The names of objects of the Document Hierarchy at five levels: **Application, Batch, Document, Page** and **Field.**

- Indications that Rulesets have been bound to certain objects in either or both nodes of an object's Ruleset tree: *In* and *Out.*

✓ ***Very important!*** A shaded ⊞ *Open* node attached to an object in the list on the left means that at least one rule from a Ruleset on the right has been bound to the object.

DStudio provides some handy ways to see the relationship between the Document Hierarchy and Ruleset/Rule combinations.

SyncView: DocHier < Ruleset Icon

The *Document Hierarchy* tab presents a **complete** list of Ruleset/Rule combinations when you:

- Highlight the **Application ID** in the *Rulesets* tab.

- Press the **SyncDocHier** icon (<) at the bottom of the separating column.

If, instead, you select a **single** Ruleset from the list and click on this icon, the *Document Hierarchy* tab will list only those objects to which you have bound rules that belong to the Ruleset.

SyncView: DocHier > Ruleset Icon

If you highlight an object that is accompanied by a shaded **+Open** field in the *Document Hierarchy* tab, and press the **SyncDocHier** icon (>), the *Rulesets* tab will list the object's Ruleset/Rule combinations. When you click on a rule, the *Rulesets* tab will list its functions – and each function's actions.

If you highlight the Application ID in the *Document Hierarchy* tab and press this icon, the *Rulesets* tab lists all available Rulesets, with their rules.

AutoSync Icons

✓ Both **SyncView** icons are one-shot tools The **AutoSync** icons, in contrast, are toggle-switches: you can keep them open as long as you like.

6.5 Task Profiles

A **Task Profile** consists of Rulesets. When Rulerunner Service (RRS) is asked to run rules, it is told what Task Profile to run when carried out by a *Taskmaster RRS* task. An application's Task Profiles are listed in Datacap Studio's *Task Profiles* tab.
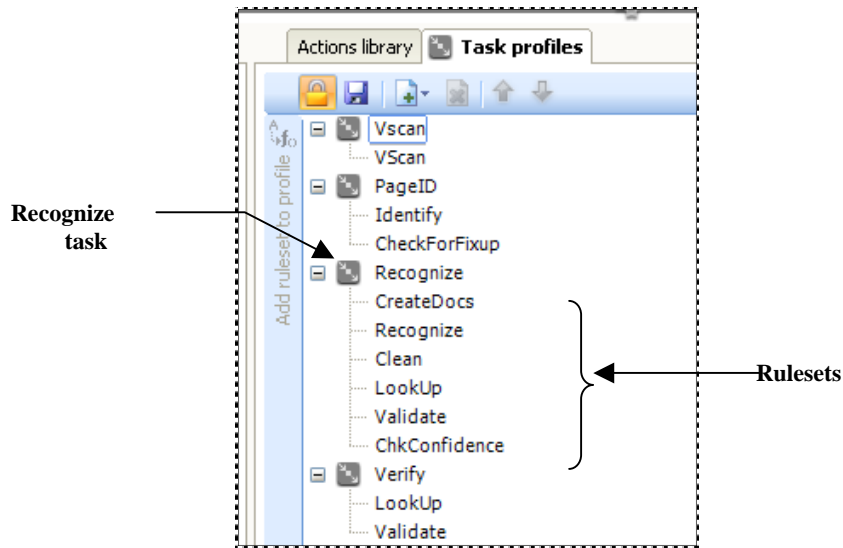

Rulemanager Page – *Task Profiles tab*

Above are the Rulesets that are part of four Task Profiles of the *1040EZ* application.

Important(1): When you assign a Ruleset to a Task Profile, you assign all its rules as well.

Important(2): The order in which the Rulesets appear in the Task Profile is the order in which the task will apply them when processing a batch and its contents.

6.5.1 Icons of the Task Profiles Tab

The icons of this tab are similar to those of other tabs:
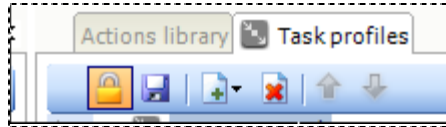
Icons of the Task Profiles tab

Lock frees up the other icons – as well as the Task Profiles and their settings.

Save updates DStudio with changes to the Task Profiles.

+ adds a new Task Profile's ID.

**X** deletes a Task Profile (but does not delete its Rulesets!)

⬆ and ⬇ move Rulesets up and down within the Task Profile.

6.5.2 How to Construct a Task Profile

✓ *Very important!* You cannot include Rulesets in a Task Profile until you have defined the applicable Ruleset/Rule combinations (Page 12).

✓ After you have organized the Rulesets that the task will apply, take the steps outlined below to put together the Task Profile.

Step	Action

1. Open the *Task Profiles* tab and toggle the **Lock** icon (!).
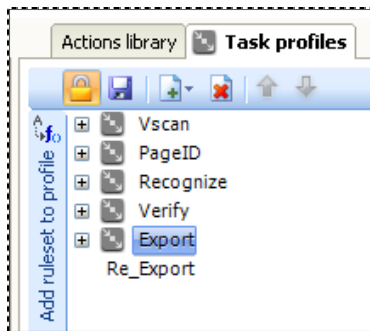
Toggle this icon.
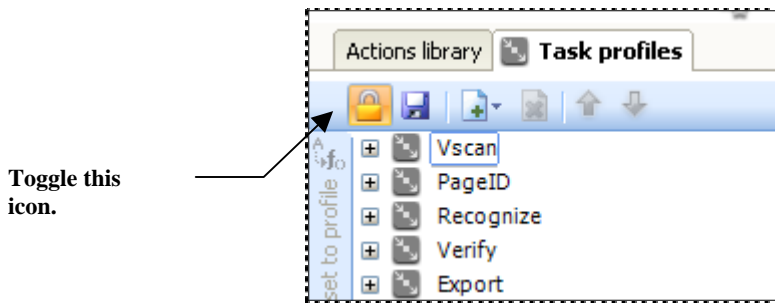
2. Press the + icon on the toolbar. Select the **Task and Job Based** option in the *New Task Profile Name* dialog, and the Job ID.
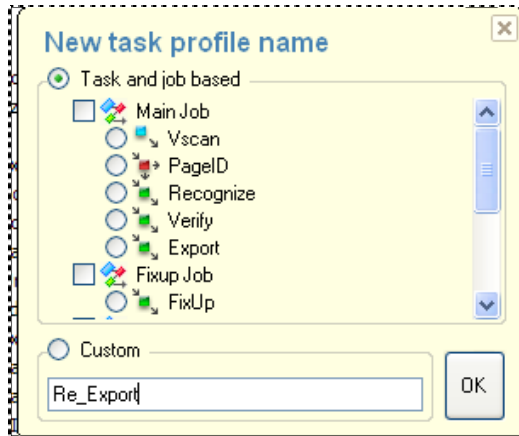
To Assemble a Task Profile (continued)

Step	Action
3.	Enter the Task Profile's name in the field at the bottom and press the OK button.
4.	Confirm that the Task Profile's name appears in the Task Profiles list.

Step	Action
5.	In the *Rulesets* tab, highlight the task's opening Ruleset and press the **Add Ruleset to Profile** icon in the *Task Profiles* tab.
6.	Confirm that the **Ruleset** is part of the task.

**Click here.**
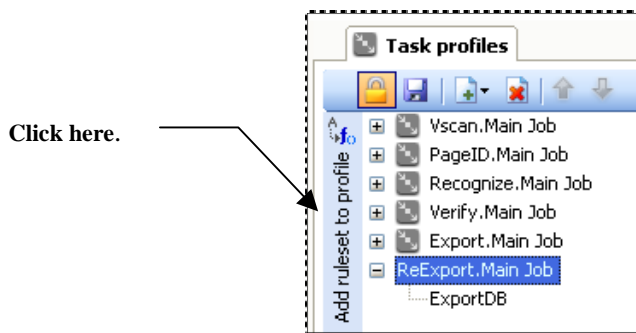
Step	Action
7.	Repeat Step # 6 to add other Rulesets to the Task Profile.

6.6 Custom Rule Execution Conditions

A typical Ruleset/Rule combination is **global.** This means that it applies to all instances of the object to which it is bound, when a task applies the rule.

In the example below, three global Ruleset/Rule combinations are bound to the *TaxpayerSSN* **Field** object. A *rulerunner* task with a profile that includes one or more will apply them every time it encounters a *source* page's *TaxpayerSSN* field.
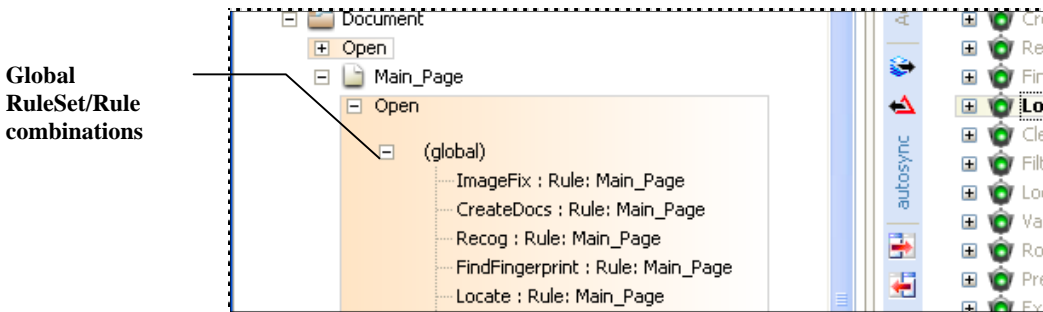
Global RuleSet/Rule combinations
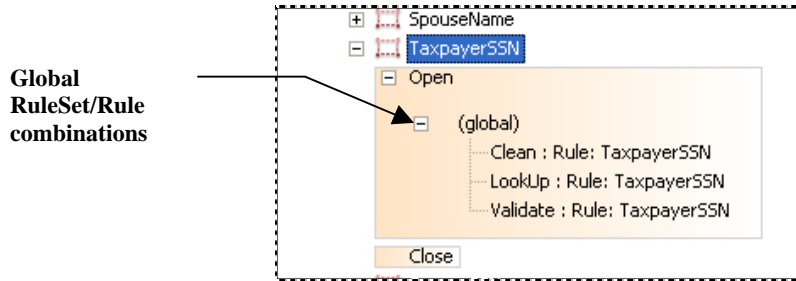
TaxpayerSSN Field Object -
Global Ruleset/Rule Combinations

✓ You can also define **custom** Ruleset/Rule combinations, which can be particularly useful if the application employs multiple fingerprints (Chapter 7), and the properties of certain objects change from fingerprint to fingerprint.

The *1040EZ* application has one fingerprint. The *Invoices* application, in contrast, has a different fingerprint for each invoice that has a unique format.

The global Ruleset/Rule combinations attached to that application's *Main_Page* **Page** object determine which pages in a batch are *source* pages; attempt to match the current *source* page to a pre-defined fingerprint; and use the fingerprint's settings to locate the fields on the page and capture the fields' data.

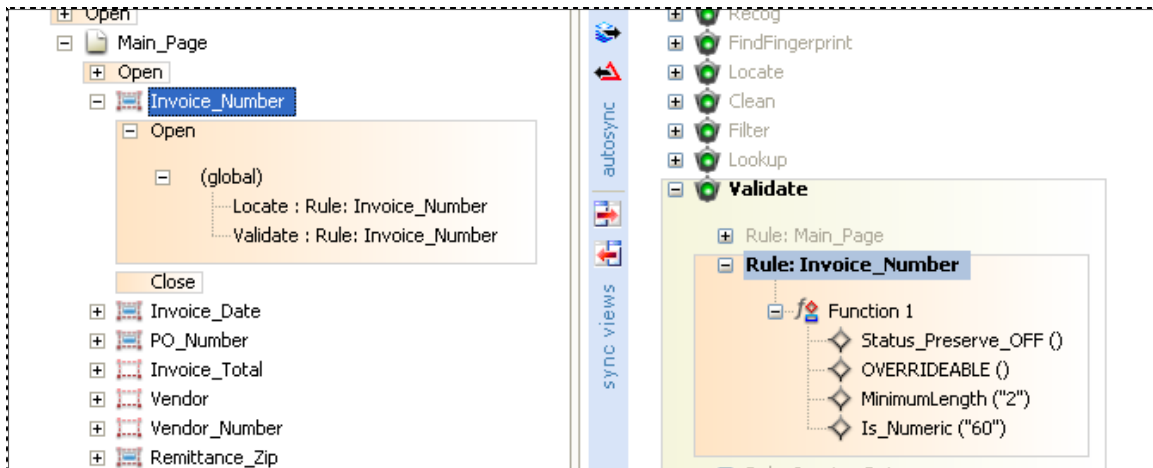**Global RuleSet/Rule combinations**

Invoices Document Hierarchy – Main_Page Object
Global Ruleset/Rule Combinations (partial list)

You can also define a custom Ruleset/Rule combination that goes into effect if the processing task encounters a specific condition associated with the bound object.

The value of an object's **STATUS** property usually indicates OK ("0") or Problem ("1"). In the default structure, however, the global **Validate** rule for *Invoice_Numbers* fields does not check for a problem status assigned to this field on any invoice.


Invoices Document Hierarchy - *Invoice_Number Field Object*
Global Validate: Rule:Invoice_Number Ruleset/Rule Combination

✓ You could define a custom rule to check for problems with *Invoice_Number* fields on pages represented by a single fingerprint – if you use procedures outlined previously as you take the steps on the next few pages.

Step 1: Assemble a Custom Ruleset/Rule Combination

Begin by putting together the Ruleset/Rule combination that will be bound to the target object of the Document Hierarchy – in this case, the *Invoice_Number* **Field** object.

Here, for example, the **OVERRIDEABLE** action of the global rule has been replaced by a **NON_OVERRIDEABLE** action.


Validate/RuleC Combination
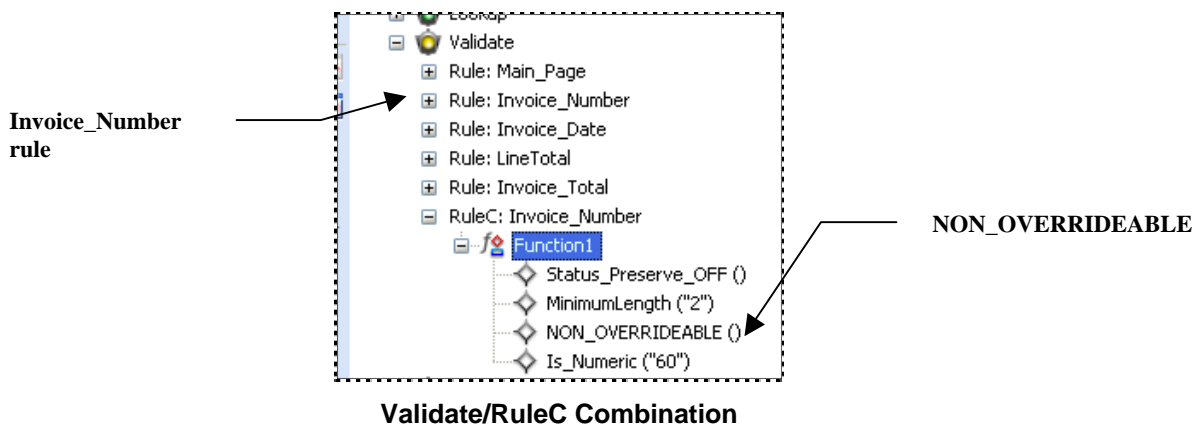
Step 2: Toggle the Lock Icon

Click the *Document Hierarchy* tab's **Lock** icon.

Step 3: Select a Condition's Focus

To formulate of a custom Ruleset/Rule combination, you must be prepared to specify a condition involving a property or variable of the object you are working with, or of another object any level of the Document Hierarchy (see the next page.)

This begins when you right-click on the object to which you will bind the custom Ruleset/Rule combination, and select **Open Condition** from the **Add** options. Follow up by selecting the *New Condition* option to access the *Custom Rule Execution Condition* dialog.

Add Options

Custom Rule Execution Condition dialog
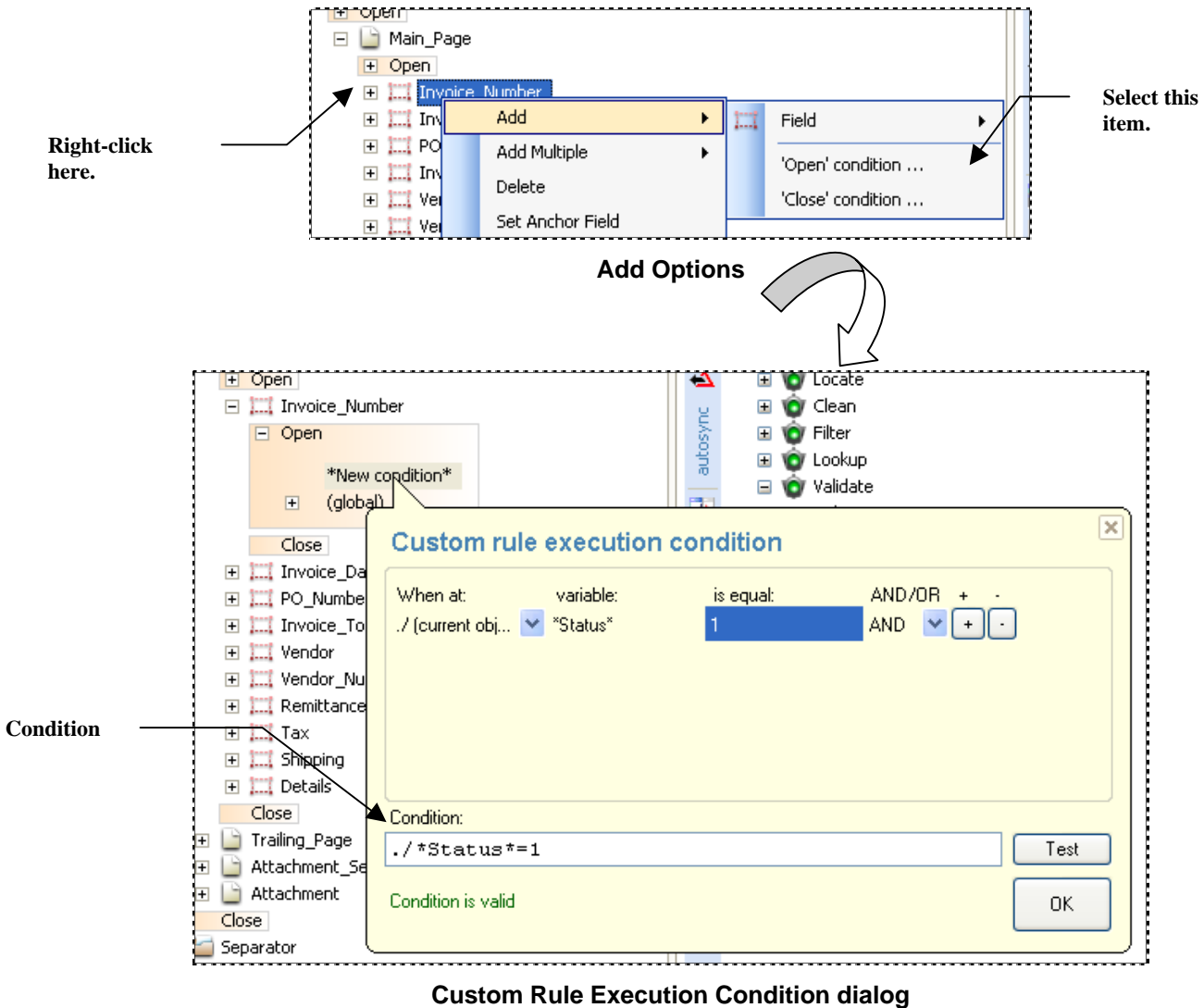
The *Custom Rule Execution Condition* dialog helps specify a condition with one or more parameters. The table on the next page describes the dialog's fields and settings.

Custom Rule Execution Condition Dialog

Field or Setting	Description
When at…	A drop-down list of possible "location" parameters for a condition, including:
	`./(current object)` The Ruleset/Rule combination will be carried out when a value of a property or variable of the selected object is equal to the amount in the **is equal** field.
	`.../(parent next level)`
	`../../(parent two levels above)`
	`./(current object)` The Ruleset/Rule combination will be carried out when a value of a property or variable of the selected object is equal to the amount in the **is equal** field.
	`../(parent next level up)`
	`../../(parent two levels up)`
	`.../(any parent)`
	`/(root Batch)`
Variable	The name of a previously established variable of the object(s) you selected from the **When at** drop-down list.
	(Chapter 5 explores properties and variables the Document Hierarchy's objects, at three levels: **Batch**, **Document**, **Page** and **Field**.)
Is equal	The value that sets the condition.
	When the variable's value is equal to the value you place in this field, the *rulerunner* task will apply the custom Ruleset/Rule combination to the object.
AND/OR + -	Adds a parameter to the condition.

Step 4: Select an Object of the Document Hierarchy

Select the object of the Document Hierarchy to which you are attaching the custom Ruleset/Rule combination.

The Test button checks to see if the custom condition can be correctly bound to the object of the Document Hierarchy that you have selected.