fDocs Actions

1   Introduction

*f*Docs Actions is an optional facility for *f*Docs that lets the document designer specify actions to be taken by the system when a document is generated. The system offers two default actions (send as email attachment and save to a server folder) but others may be added by the creation of plugin components for specialized purposes (e.g. to save to a document management system).

The actions available are offered to the document designer as special tags in the *f*Docs Designer. The tags can be placed anywhere in the document (they will be removed when a document is generated) and they include the instructions for the action to be taken.

Any number of action tags may be included in a document and since they are *f*Docs tags they can be included in data group loops (to take the action for each record in a data group) and within condition and compare tags to make the action conditional on any factor the designer chooses.

2   Designing the Template

All actions that are available to the document designer are shown in the Special Tags section of the *f*Docs Designer Word add-in.

In this example there are two options: email the generated document and save the generated document to the server.
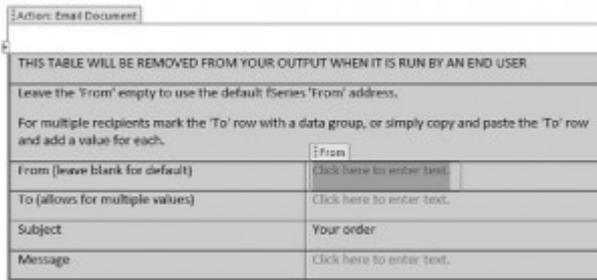
More options can be added in *f*Admin.

Double click on an action to place it into the document. It doesn't matter where you place it as the tag will be completely removed when the document is generated (although it will show with parameters filled in when previewing the document).
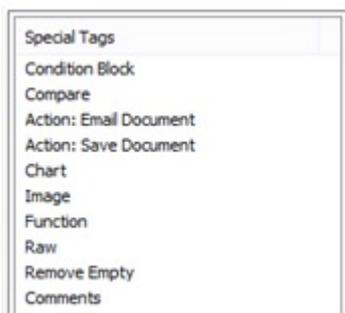
2.1   Setting the Action Parameters

The tag in the document includes some information about the action and a table of parameters that are required.

Each row in the table represents a parameter that will be used to take the action. The right column contains an area into which the parameter must be entered. It is essential that the value (or tag) is placed inside the control, otherwise the value will not be found by the system when executing the action. In this example the 'From' parameter entry is highlighted showing the control into which the value must be placed.

Each parameter value may be made up of directly entered text or *f*Docs tags, in fact anything that may be placed in a template may be placed inside the value tag. So in this case the 'From' value would probably be the Email tag from the data being used to generate the document.

It is also possible to have multiple values, for example if you had a data group for team members you could include a data group loop in the 'From' value to include the email address of each member (separated with a semi-colon) to send the email to all members.

You can also use condition and compare tags inside value tags so that the value is conditional. For example the message may be a detailed document in its own right with sections that are dependent on the document's content.

2.2   Multiple Actions

Any number of action tags may be inserted into a template. Each will be evaluated and executed separately.

You can even include action tags in a data group loop so that the action is executed for each record in a data group. For example, if you had a data group for team members, you could put the 'Email Document' action in a loop for this data group and have the email sent separately to each team member.

2.3   Conditional Actions

By surrounding an action tag with condition and compare tags you can make the actions conditional.

For example you may only want to send the email if you have an email address to send to.

Another example is if you wanted to take slightly different actions according to different circumstances. For example, if the message was quite different depending on whether the order was for a one product type or another. You could insert different action tags for each email format and then put condition/compare tags around each action to ensure only the appropriate email was sent.

3   Testing the Actions

When previewing the document the actions are not executed but the action tags are left in place but with the parameter values evaluated as they will be when generated by the end user.

This allows you to check that the action will be taken correctly without actually taking the action.

4   Configuring the Available Actions
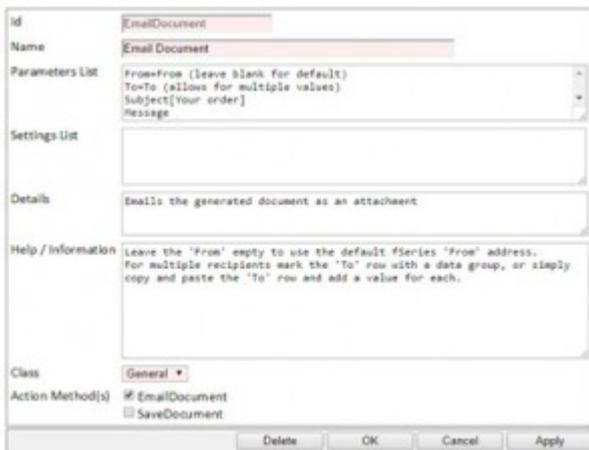
Each available action that appears in the Special Tags list is a configured option in the *f*Docs Actions section of *f*Admin.

This example shows the configuration for the Email Document example action tag.

The **Parameters List** is the list of parameter values that you want the document designer to fill out for the action.

Each item in the list represents one parameter in the form, It is specified in the form:

Id[Default value]=Label text

The Id uniquely identifies the parameter as required by the action. The default value (in square brackets) is optional and will be inserted as a default parameter value. The Label text is optional (the Id is used if not specified) and is shown in the left side of the parameter table.

The **Settings List** is similar to the parameters list except that the values are preset and may not be specified by the template designer. For example,

when saving a document you may wish to specify here exactly which folder the document is to be saved to. The format is id=value, one per line.

Depending on the action method, parameters and settings may be used for the same value. This makes it possible for the administrator to choose to allow the designer to specify a parameter value but have it substituted with a setting if no value is available when the document is generated.

The **Details** text is a note for your own information to described the action.

The **Help / Information** is the text shown at the top of the action tag to assist the template designer in specifying the action.

Select a 'Class' for methods to be offered. Select a method to determine the actual action to be taken. A class is a plugin component that contains one or more action methods. The system includes the 'General' class which include the 'Email Document' and 'Save Document' methods

Note that any method may be used in any number of actions. So for example, you may wish to offer the template designer different ways to save documents (e.g. into different locations). Set up a different action for each, but using the same class/method. However, take care not to overwhelm the designer with too many options.