

7. Workflow

7.1. Introduction

Workflow commonly refers to the automated routing of tasks and documents, and it's a powerful tool that can quickly help a well-managed business run more efficiently, streamlining and speeding up fundamental business processes.

There are several ways to implement workflow in dbFramework:

- **Process workflow, using the workflow designer** (this section)
Enables you to graphically design the routing and stages of tasks, known as process workflow or business process modeling. An example of this type of workflow is a support ticket. This starts with a form where the customer enters the details of the problem. Once submitted it is screened by the general support technician where it is either answered immediately, or escalated to an expert. The expert will then optionally request more information, to which the customer will respond, and finally the support ticket will be resolved.
- **Approval workflow** (see section 11.2)
Lets you specify areas of the database that cannot be entered or modified without approval by a specific user group, optionally with conditions. An example of this type of workflow would be a purchase request. The request cannot be entered without approval by the supervisor. If the purchase is for more than \$5k, for example, it would also have to be approved by the department manager.
- **Custom actions and event handlers** (see section 8)
A highly extensible framework for adding custom actions and event handlers using programming languages. For workflow that involves complex decisions involving advanced calculations and data aggregation, this type of scripting can be employed.
- **Integration with external systems** (see section 21)
The ability to interact and integrate with external systems using the XML API. For example, workflow may be started by an event in a separate system. Or some stage in a workflow may involve triggering an action in an external system.

This section will detail the workflow designer features of dbFramework.

Note: The screenshots in this section are based on the WorkflowFirst product. All options are the same as those in dbFramework. WorkflowFirst provides a *subset* of the functionality available in dbFramework.

7.2. Workflow Basics

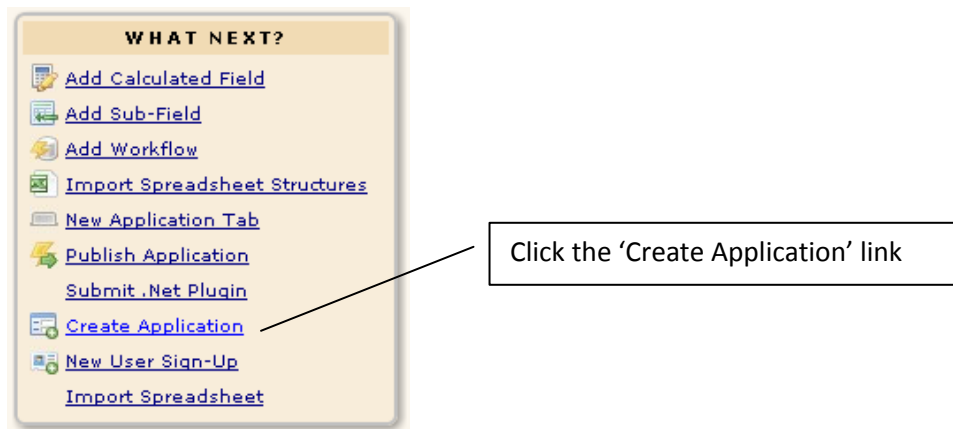
Workflow can be added to any area of the database, and any area of the hierarchical data model. This means that you can have workflow available on an Invoice itself, on the Invoice Items or on the Customer record.

Just like anything created in dbFramework or WorkflowFirst, you first design the workflow and then, when you want to *use it*, you must click 'Publish Application' and run the application in another window.

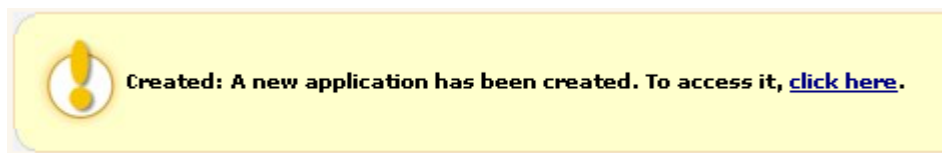
Creating workflow using dbFramework or WorkflowFirst is far simpler than other business process modeling systems because you don't have to design any forms or databases, instead it creates these for you as you go. It lets you focus on the workflow itself instead of having to design forms.

7.2.1. Creating a New Application

If you're starting a new workflow application, click the 'Create Application' link in the tasks list:



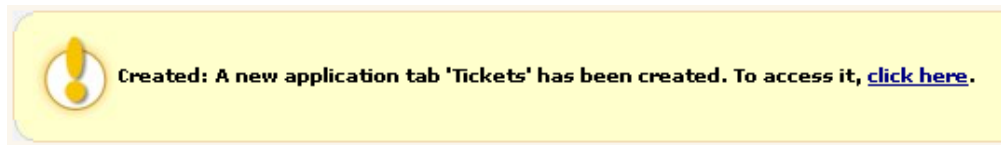
Once created, it will display a link like this:



Click that link to take you to the application record. Next you need to add an application tab. This also adds a record type. To do this click the 'New Application Tab' button:



Once you create the application tab you'll see a message like this:



Click the link to take you to the type. For an existing type that you created previously, just go to the type record where you wish to add the workflow.

7.2.2. Adding the Workflow

An example of an existing type definition is shown in Figure 20. You can then either go to the Workflows drilldown (go-to item) and click 'Create New', or you can click the 'Add Workflow' button.

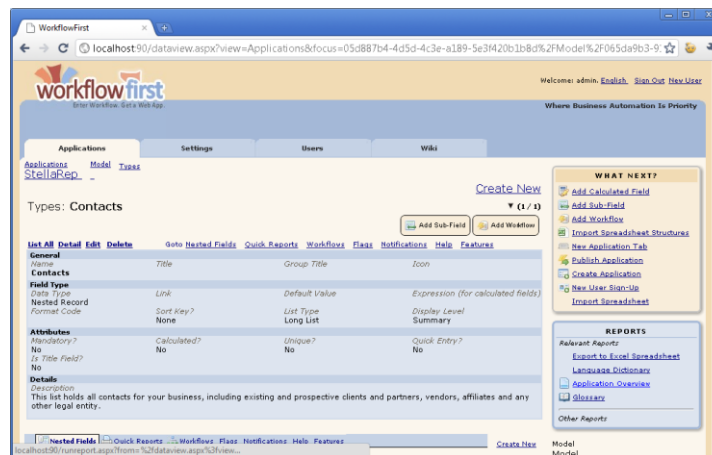


Figure 20: Type definition screen, showing workflow button



Figure 21: The 'Add Workflow' button

Clicking the 'Add Workflow' button (or clicking Create New on the workflow list) will display a form where you can enter the name of the workflow.

The screenshot shows a form titled "Add Workflow: Workflows". It contains several input fields and dropdown menus. The "Title" field is empty. The "Workflow Type" dropdown is set to "Updates Existing Item". The "Condition" field is empty. The "Show State" dropdown is set to "Yes". The "Create Report" dropdown is set to "No". The "Email Notification" dropdown is set to "No". The "Assignable" dropdown is set to "No". The "Audit Trail" dropdown is set to "Yes". The "Locked?" dropdown is set to "No". There is a large text area for "Notes" which is currently empty. A "Create" button is located at the bottom right of the form.

Figure 22: Adding a new workflow

The following fields can be entered:

Title

The name of the workflow, which is just used for internal reference purposes and isn't displayed to the user. For example, "Help Desk Ticket System".

Workflow Type

Either 'Updates Existing Item' or 'Creates New Item'. This refers to how the workflow starts.

- **Updates Existing Item:** The workflow can be started once a record has already been created. It will modify (update) the existing record as it goes.
- **Creates New Item:** The workflow can be started *before* a record has been created. The workflow creates a new record and then updates it as the stages progress.

Condition

This is an optional, simple condition that is evaluated to determine if the workflow start button should be visible. For example you could put 'Complete=False' here to ensure that the first workflow stage will only be visible if the Complete field is set to False.

Show State

Indicates if the current workflow stage of the record should be visible as a field.

Create Report

Indicates if a workflow report should be provided. This will provide a list of all records in the workflow, grouped by their current stage. This report can be subscribed to so that it is automatically emailed to your inbox each day.

Email Notification

Indicates if a Notifications area will be added, letting users subscribe for email alerts when records reach specified workflow stages. The email will include a link to the workflow record so that further actions can be taken.

Assignable

If this is set, then the record will be turned into a task. Various fields will be added related to tasks, such as progress, estimated completion dates and a user assignment field. These fields can be selected as input fields when creating the workflow stage. When the 'Assigned To' field is set, the system will automatically send an email to that user to tell them that a task has been assigned to them. They can then click the included link and participate in the workflow.

Audit Trail

Indicates if a history will be recorded of all changes made to the record, whether through workflow or by manual editing. An 'Audit Trail' drilldown will be added and the system will automatically create an entry in this list whenever a change is made to the record.

Locked

Indicates if you wish to lock the workflow so that it won't be updated by any upgrade process. This is only applicable if you are customizing a sample application.

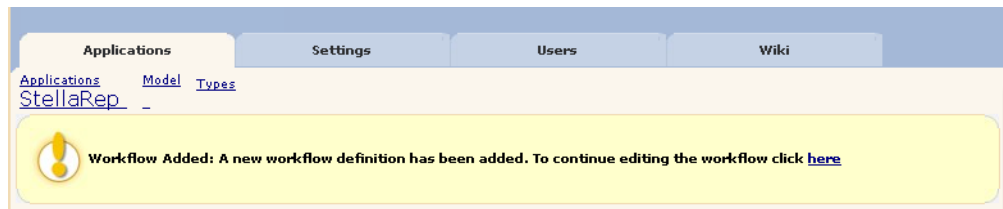
Notes

This is an area for entering in any notes you want to make on the workflow design or purpose. These are not shown to the user and are used for internal reference purposes.

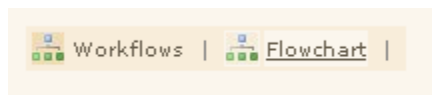
Once the workflow is created, it will display a message with a link that will take you to the workflow designer.

Note: Any number of workflows can be added to a field, and any number of workflows can be in progress at any time. Because they share access to the same record,

workflows can also interact with one another, allowing you to model some very sophisticated business processes.



You can also click on the 'Workflows' drilldown, select the workflow you just created, and once that's displayed click the 'Flowchart' view:



...to switch to the designer. The 'workflows' view lets you edit the data of the workflow without the graphical designer to assist you. This may be useful for certain operations like rearranging stages.

Once the flowchart is displayed, the initial screen will just show the main workflow starting point:

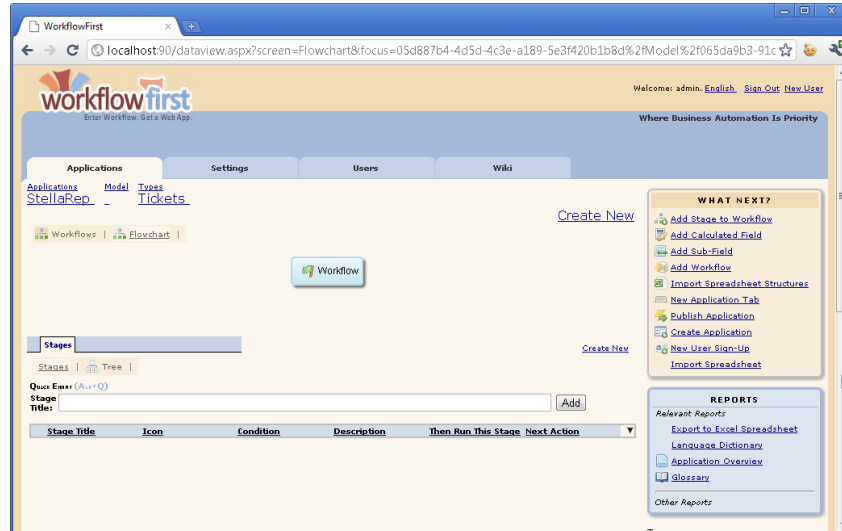


Figure 23: Initial view of the workflow designer

If you put your mouse cursor inside the blue box, a number of toolbar options will appear:

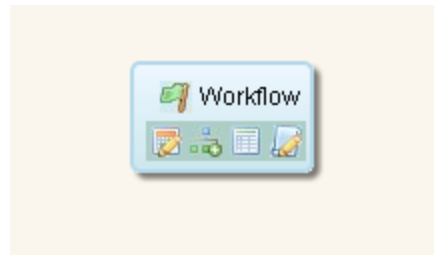


Figure 24: Action buttons available from within a workflow item

From left to right, the icon buttons are:

Edit

This command takes you to the edit screen for this workflow stage (or the main workflow record). Here you can edit the workflow stage title, icon and other setting.

Add Sub-Stage

This command takes you to the 'New Sub-Stage' form where you can enter the details of a new stage that will go under this stage. We'll be covering this next.

Select Input Fields

For a workflow stage that has user-input, this lets you select the fields that will appear in the form generated for that step in the workflow. This can consist of existing fields, or new fields you can specify as you create the stage. This will be covered in this section.

Edit Script

In addition to letting the user enter fields and allowing the user to make decisions, you can also add some ad-hoc scripting. The scripting will have complete control over what can be set in the stage and also provides the ability to easily start a secondary workflow. Details of the scripting language, DbfScript, are provided in section 20.

7.2.3. Workflow Stages

7.2.4. Concepts

The next step is to add some workflow stages. These are displayed in the workflow designer, and the process flows from top to bottom. An example of a workflow in the designer is shown below:

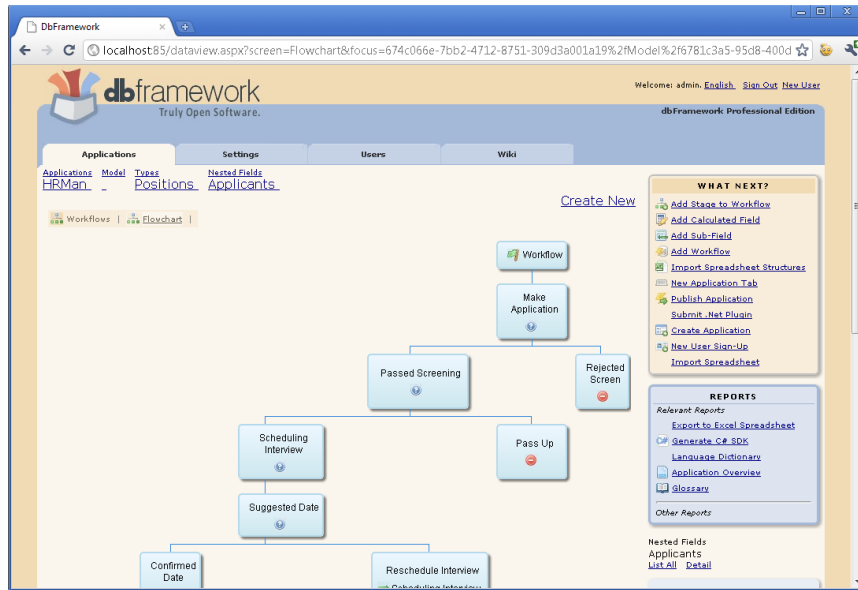
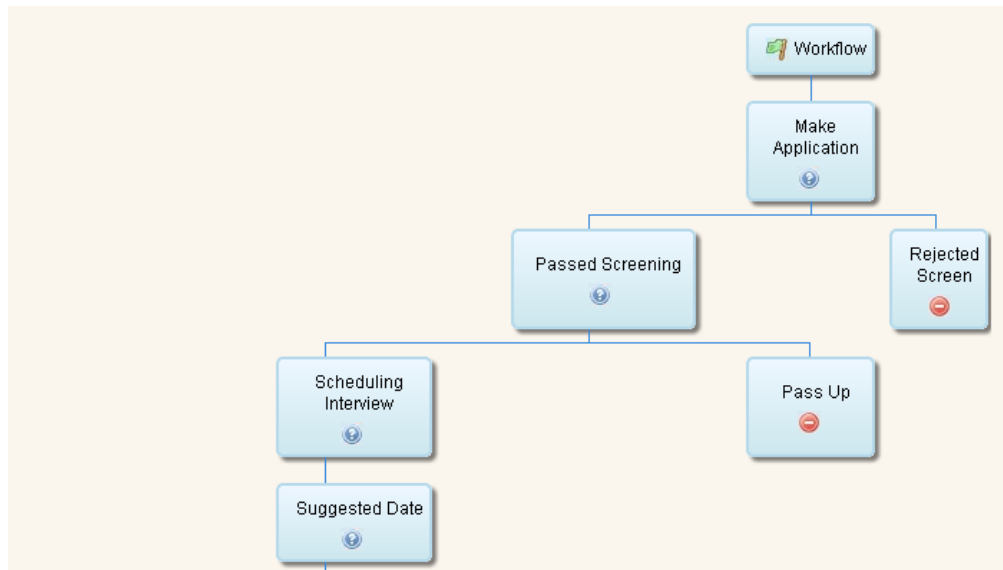


Figure 25: Example workflow in the designer

We'll look more closely at this particular workflow to better understand how the structures relate to the business processes. In this case we're modeling the process of applying for a vacant position at a company.



The top most box in the flowchart is the Workflow Record, and every workflow has one of these.

Underneath that are all the stages. The 'Make Application' box is the first stage. This collects information from the user about the position application. Once published in the application, this translates into an action and a form.

Positions: Accounting Clerk (1 / 1)

Change Filled Status Make Application

List All Detail Edit Delete Goto Skills Applicants Notifications Audit Trail Report Subscriptions

Title	Filled
Accounting Clerk	No
Notes	

Skills Applicants Notifications Audit Trail Report Subscriptions Create New

Quick Entry (Alt+Q)

Skill: Add

Skill Critical

Figure 26: The 'Make Application' button appears automatically on a Position record

hrmanager Access Your Company

Welcome: admin English Sign Out New User Search

Human Resources Manager

Department Positions Skills Interview Guides Org Chart Users Configuration Tag Navigator

Make Application

Make Application

Full Name Terry Greenfield EMail tgreen@gmail.com

Address 5 Rossmore Road, Stony Point, NY 10983

Telephone 845 512 2156

Resume Attachment Choose File my_resume.docx

Create

© Copyright RiaForm Technology 2011

HRMan Server: 0.1.125 (beta) Up Since 3/27/2011 4:2:40:5 PM

Powered by RiaForm dbFramework

Figure 27: Once clicked, an empty form will appear with the fields you need to enter

After the user fills out the 'Make Application' form and submits that, it automatically creates a new record in the 'Applicants' drilldown from the Position record.

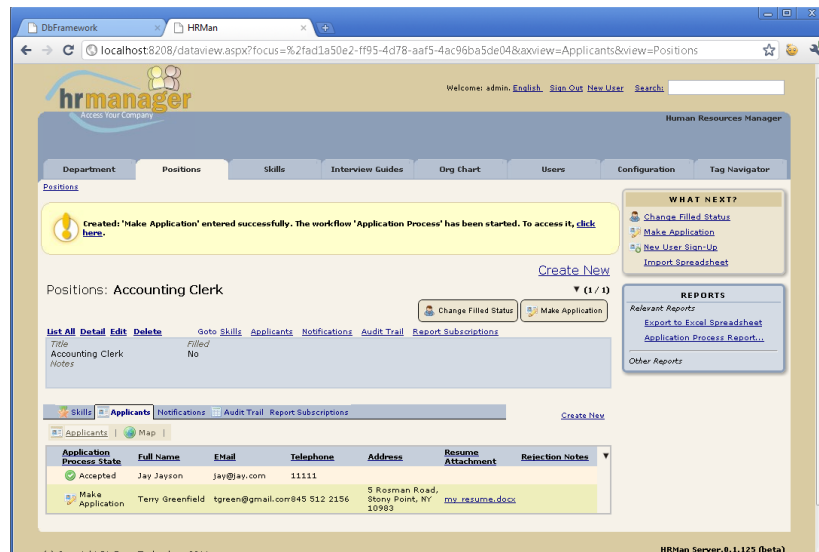


Figure 28: Once submitted, the new record appears at the bottom, and a message at the top.

Anyone subscribing to notifications (we'll touch on this in a later section) will receive an email telling them that a new Applicant record has been added, and when they click on that link they'll be taken to the record.

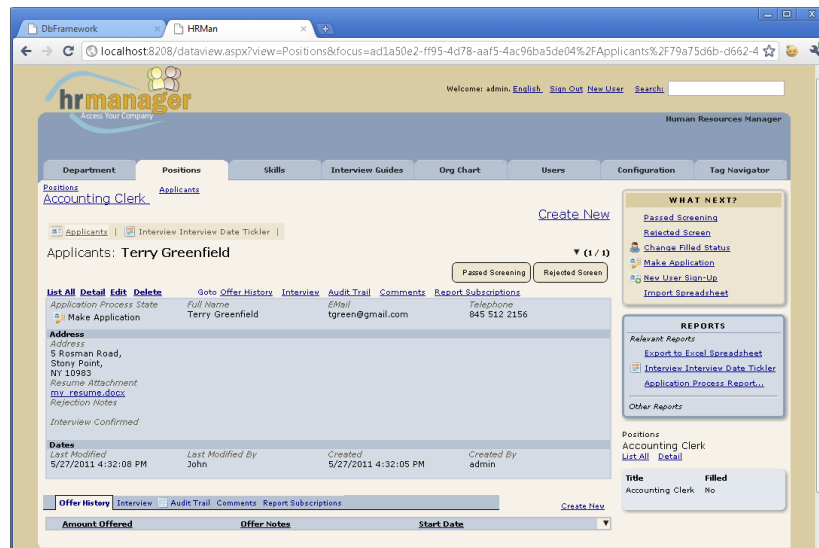


Figure 29: The newly created Applicant record, with action buttons at the top

Now, getting back to the workflow designer in Figure 25. Underneath the 'Make Application' stage, processing splits between two further stages: "Passed Screening" and "Rejected Screen". It's those two stages that correspond to the buttons you'll now see above the new applicant record. By adding those sub-stages, we've added a choice for the user to make.

Once they click one of those buttons, workflow processing will continue underneath that stage.

So in this case we may click 'Passed Screening'. Refer back to the flowchart and you should see what options this will now provide on the record:

Applicants: Terry Greenfield (1 / 1)

[Scheduling Interview](#) [Pass Up](#)

[List All](#) [Detail](#) [Edit](#) [Delete](#) [Goto Offer History](#) [Interview](#) [Audit Trail](#) [Comments](#) [Report Subscriptions](#)

Application Process State	Full Name	Email	Telephone
Passed Screening	Terry Greenfield	tgreen@gmail.com	845 512 2156

We now see the options of 'Scheduling Interview' and 'Pass Up'. If we click 'Scheduling Interview', this can then be sent to whomever schedules interviews – the secretary for example. Again refer to the flowchart and you'll see the option 'Suggested Date' under that. Once we click that button we get a form like this:

Suggested Date

Interview Date: Interviewer:

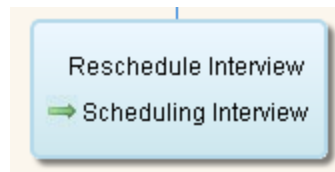
Calendar: May 2011. Today: 27 May 2011. The date 27 is highlighted in red.

[Create](#)

As we enter this data in, it automatically populates the fields in the Applicant record. At the end of the workflow we'll have a complete record. Because WorkflowFirst and dbFramework are hierarchical, this record can contain some pretty complex structures, enabling some very sophisticated workflow to be modeled.

Lets continue to look at the rest of the workflow for the Position Applicant, continuing on from 'Suggested Date' for the interview, in Figure 30.

The 'Suggested Date' stage forks into two stages, 'Confirmed Date' and 'Reschedule Interview'. We'll look at the 'Reschedule Interview' stage because, if you look closely, you'll notice a green arrow at the bottom of the box:



This means that if the 'Reschedule Interview' option is selected, then the workflow will jump back to the 'Scheduling Interview' stage, going back to showing the 'Suggested Date' option where you can enter in a new date. This is controlled by selecting the stage in the 'Then Run This Stage' option for that stage.

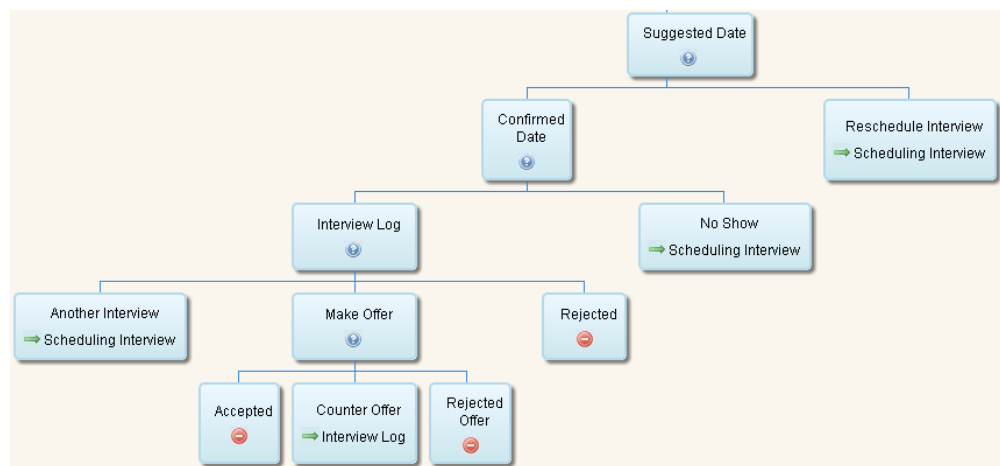


Figure 30: The remaining of the Position Applicant workflow

We use the same technique in the 'Another Interview' stage, so that we can schedule a new interview and *re-use* the processes involved in that.

Now this brings up an interesting issue: we want to store a history of each interview that takes place, we don't want it to overwrite the interview details for each one that is scheduled. How can we do this?

When you add a field to a stage, which goes to constructing the form the user will fill out, you have an option to specify a 'Separate List Tag'. This lets you put the entered value into a separate sub-list of the record. It appends values to that list, so that every time that stage is reached a new entry is made in that sub-list. This lets you construct highly structured data in one simple workflow.

The final concept worth mentioning in this section are the terminators.

For each stage, you have to specify what happens next. This is put into the 'Next Action' dropdown at the end of the stage. This is what controls the icon that appears in the workflow stage box. This can be:

User Decision

This means that the next stages will appear as buttons on the record when this stage is reached.

Auto Decision

This means that the next stages will be evaluated automatically. The Condition on the stage will hold an express that the system will automatically evaluate to determine which path to take. No user action is required.

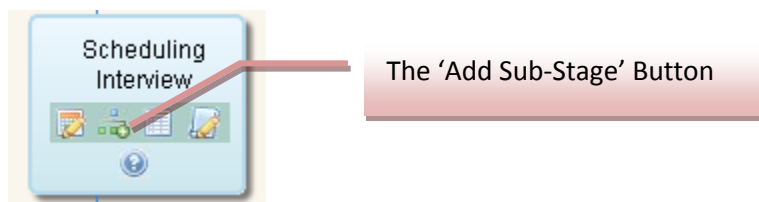
Finish

This means that the workflow is complete and no further actions can be taken.

If the 'Then Run This Stage' is specified, then the above options are ignored.

7.2.5. Adding a Stage

To add a new stage, you hover your cursor over a box in the flowchart diagram, and select the second button 'Add Sub-Stage':



This will bring up a new stage form for you to fill out:

Sub-Stages ▼

<i>Stage Title</i> <input type="text"/>	<i>Icon</i> <input type="text"/>	<i>Condition</i> <input type="text"/>
<i>Description</i> <div style="border: 1px solid #ccc; height: 60px; width: 100%;"></div>		
<u>Add a 'Fields To Collect' entry</u>		
<i>Then Run This Stage</i> <input type="text"/>	<i>Next Action</i> <input type="text" value="User Input"/>	

The following fields can be entered:

Stage Title

The title of the stage is used to title the action button, and is also used for internal reference purposes.

Icon

An optional icon for this stage, that will be displayed in the task name and action buttons.

Condition

For automatic decisions, a DbfScript expression that will be used to determine whether this stage will be selected. If multiple stages evaluate to true, then the last stage will be selected.

For 'User Decision' stages, the Condition field holds a simple expression that defines whether the option will be visible to the user. This works the same way as the Visible Condition, described in section 6.9.1. It can be used to effectively suspend a workflow until a certain condition is met, often triggered by a secondary workflow or process.

Description

The description of the stage. This is used for reference purposes and isn't displayed to the user.

'Fields to Collect'

Click this to add a field reference or a new field to use in generating the input form for this stage. This is discussed below.

Then Run This Stage

Optional. If you wish this stage to jump to another stage on completion, then select the stage to jump to here.

Next Action

Defines how the next set of stages (the sub-stages) should be processed. If 'Auto Decision' is selected, then the conditions of each sub-stage is used to determine if that stage should be followed. If 'User Decision' is selected then the sub-stages are displayed as buttons presented to the user.

Once you are done entering the details of the stage, click 'Create'. This should take you back to the flowchart where you can continue adding additional stages.

7.2.6. Adding Fields to Collect

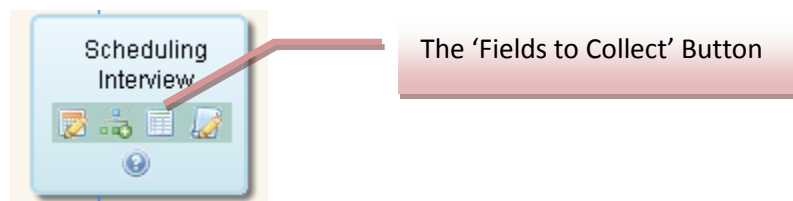
Any stage in the workflow can collect information from the user. In many ways the workflow features in dbFramework and WorkflowFirst focus on assisting in a process of gradually filling out a form.

There are several ways to specify the fields that should be collected at a particular stage.

When you are entering a new stage, you will see a 'Add 'Fields to Collect' Item' link which you can click for each field you want to add.

[Add a 'Fields To Collect' entry](#)

Alternatively, from the flowchart you can access the fields associated with that stage by clicking the third toolbar button:



From that list, you can then click 'Create New' to add a new field.

Lastly, from the stage record view, you can select the 'Fields to Collect' drilldown to get to the same list.

Once you are on the form for adding a new field to collect, you will see something like this:

<i>Field Reference</i> <input type="text"/>	<i>New Field Title</i> <input type="text"/>	<i>New Field Type</i> Text (Few Words) ▾	<i>Is Mandatory</i> No ▾
<i>Ask User</i> Yes ▾	<i>Value To Use</i> <input type="text"/>	<i>New Group Title</i> <input type="text"/>	<i>Separate List Tag</i> <input type="text"/>

The following fields can be entered:

Field Reference

Here can you select an existing, previously created field in this record, or in a sub-record.

New Field Title

Alternatively, just enter in a new field title here. If the field doesn't exist, it will be created for you automatically when the stage is created, and it'll then be available to select as a 'Field Reference'. Also, the next time you view this entry, you'll notice that the field will be selected in the 'Field Reference' automatically.

After creating the stage, you can also click on the Field Reference link to go to the actual field definition, where you will find a multitude of additional options to modify the behavior of the field. See section 6.6 for more information.

Tip: The ability to create the data model as you are entering the workflow saves a lot of time and makes creating workflow that much more efficient.

New Field Type

If the previous entry was used to create a new field, then you can specify the data type (that is, the type of data it will hold) using this dropdown. You can review the different data types in section 6.6.3.

Is Mandatory

Specifies whether this field will be required to be entered by the user. If this is set and the user does not enter this field, an error will be displayed.

Ask User

Specifies whether this field should be included in a form so that the user can enter the value. If not, then the next field will be used to seed the value for the field, instead of it being asked of the user.

Value To Use

The value to assign to this field if 'Ask User' is set to No.

New Group Title

Allows you to specify a field group title for this field, starting a new field group.

Separate List Tag

Allows you to record this field value in a sub-record. This is a powerful feature and is discussed in more detail below.

7.2.7. Making Sub-Lists

Normally, when the workflow type is set to 'Creates New Item', the workflow system will create just one record. Then each stage can fill-in different fields in that record as the workflow progresses.

However there are times when you want to record the same field more than once, and want to keep a record of each value that was entered. To do this, we introduced the 'Separate List Tag' field.

By entering a name in the 'Separate List Tag' field, you are telling the system to store the value for this field in a sub-record of that name. You can store multiple fields in that sub-record. Each time the stage is reached (including when it is jumped back-to) a new record will be added to that sub-record list, and the values transferred into that entry.

Example: Adding Notes

Let's look at the previous workflow where we were dealing with a job applicant. At some point in the workflow we are arranging and performing interviews. Let's say we wanted to record some notes for that interview.

Now that interview can repeat 2 or 3 times, for when we want a second or third interview. Normally, if we just added a field called 'Notes' and set that, each time we interview someone that field will be overwritten. Rather, we want to add those notes to a list.

So, to do this, we add a field called Result, but also specify a separate list tag of 'Interview', like this:

The screenshot shows a configuration form for a new field. The fields are:

- Field Reference:** (empty)
- New Field Title:** Result
- New Field Type:** Text (Few Words)
- Is Mandatory:** No
- Ask User:** Yes
- Value To Use:** (empty)
- New Group Title:** (empty)
- Separate List Tag:** Interview

 A 'Create' button is located at the bottom right of the form.

Figure 31: Specifying the separate list tag

Now, when we publish the application and come to enter in the details of the interview, in our workflow form we'll see a simple field called 'Result':

The screenshot shows a form titled 'Interview Log'. Inside, there is a field labeled 'Result' containing the text:


```
The interview went very well. He had great communication skills and answered all of the interview questions correctly and was very enthusiastic about our products. I definitely recommend him.
```

 A 'Create' button is located at the bottom right of the form.

But once the record is created, it puts this value into a sub-list of the record:

The screenshot displays a record for an interview. At the top, it shows 'Interview Confirmed' with a 'Yes' status. Below this is a 'Dates' section with a table of metadata:

Last Modified	Last Modified By	Created	Created By
5/27/2011 11:39:48 PM	admin	5/27/2011 4:32:05 PM	admin

Below the metadata is a navigation bar with tabs for 'Offer History', 'Interview', 'Audit Trail', 'Comments', and 'Report Subscriptions'. A 'Create New' link is visible on the right. Underneath, there is a sub-list for 'Interview' with a 'Calendar' icon. The sub-list contains one entry:

Interview Date	Interviewer	Result
5/26/2011	John	The interview went very well. He had great communi...

Figure 32: Showing Applicant record, with the Interview sub-list at the bottom

There is no limit to the number of sub-lists that can be created under a record. And even those sub-lists can themselves have sub-lists, and so on.

You can also select a record field, instead of a simple field, in the 'Field Reference' list and this will add a link to a sub-form in your workflow form. This will let the user fill out the entire record, and that entire record will be added to the sub-list.

7.2.8. Publishing the Application

To see your workflow application, you'll need to click Publish. The 'Publish Application' link is available from anywhere in dbFramework or WorkflowFirst in the 'What Next?' list. Once you click that (and click OK) it will create your application for you, and give you a link you can click to see the application.

You'll usually find the task that starts the workflow in the 'What Next' list.

7.2.9. Adding Script

If the user decision making and conditional decision making is not enough, you can also add script to any stage. This script will be executed once the stage is complete.

Scripts can be used to make certain calculations or decisions that are hard to describe in the flowchart design. They can also be used to start secondary, parallel workflows by creating records in other parts of the database that have workflow associated with them.

In the script, the following variables are preset and available for use:

#input

The #input variable holds the current record associated with this stage. It can be modified and then updated in the database using the save command.

&path

The `&path` variable holds the path to the current location in the database.

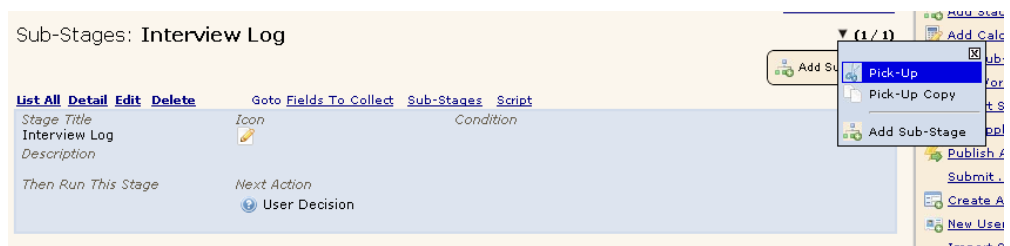
For more information on DbfScript, see section 20.

7.2.10. Deleting or Moving a Stage

At some point you may want to delete a stage. Keep in mind that when you delete a workflow stage you are also deleting all the sub-stages below it, along with all the field references. If you want to salvage some of the sub-stages, you should first **move** those stages to a different branch that won't be deleted.

To delete a stage, click the title of the stage in the flowchart diagram. This will take you to the stage record. Then click the 'Delete' link on the far left.

To move a stage, use the dropdown clipboard menu:



Select 'Pick-Up' to move the item, or 'Pick-Up Copy' to copy the stage to another area. Once it is picked up, you should go to another area of the workflow and drop it there. To do this, follow the procedure to add a sub-stage, but click the dropdown arrow in the top right corner and select 'Drop In List'. Then click the back arrow to see the changes.

7.3. Workflow Features

7.3.1. Combining Workflows

There's no limit to how many workflows you can define on a particular record, or how many types of records you can create in your data model. This opens the possibility for modeling vastly complex systems using dbFramework and WorkflowFirst.

It's therefore possible that you will need to combine two or more workflows. By combining workflows I am referring to the ability for one workflow to spawn another, secondary workflow. It would then either wait for that workflow to complete before resuming, or will continue in parallel.

There are a few ways you can do this.

The first involves spawning a separate workflow. The original workflow will continue running in parallel with the secondary workflow. By using the 'Separate List Tag' feature to create a sub-list, you can then go to the sub-list field and add a new workflow there. The workflow type will be set to 'Updates Existing Record'. This means that as soon as the record in the sub-list is created, that workflow will be started.

The second method is to suspend the workflow at a certain point until a specific condition is met, such as a field value being set. An external process or workflow would then set that value when it completes, using a simple script, which would then cause the workflow to resume. You can utilize the 'Condition' field in the workflow stage to accomplish this.

7.3.2. Notifications and Alerts

To make designing the workflow easier, the email notifications in WorkflowFirst are handled in the resultant application, rather than in the designer. When the workflow is created it provides an area in the resultant application where users can sign-up to receive notifications for particular stages of a workflow. This works much better because it means the application design doesn't have to be changed when the users participating in the workflow changes.

The 'Notifications' is added automatically as a drill-down from the parent record of the record where the workflow is operating. If the record is at the top level of an application tab, the notifications area will be added in the Configuration tab.

List All	Detail	Edit	Delete	Goto	Skills	Applicants	Notifications	Audit Trail	Report Subscriptions
<i>Title</i>	Accounting Clerk			<i>Filled</i>			No		
<i>Notes</i>									

Figure 33: A record showing the 'Notifications' goto item (drilldown).

Once you add the notification, you'll first get to select the user that will be subscribing to the notification, and in the 'Event' list you'll see a list of workflow stages, as shown below.

Each stage in the workflow is shown as an event to subscribe to. This means, for example, if you were the CFO you may want to be notified of a 'Make Offer' stage so you can monitor the amount being offered to make sure it fits into the budget.

More likely you would be the head of the department who would be responsible for making the offer itself.

For more information no Notifications, please refer to section 8.6.

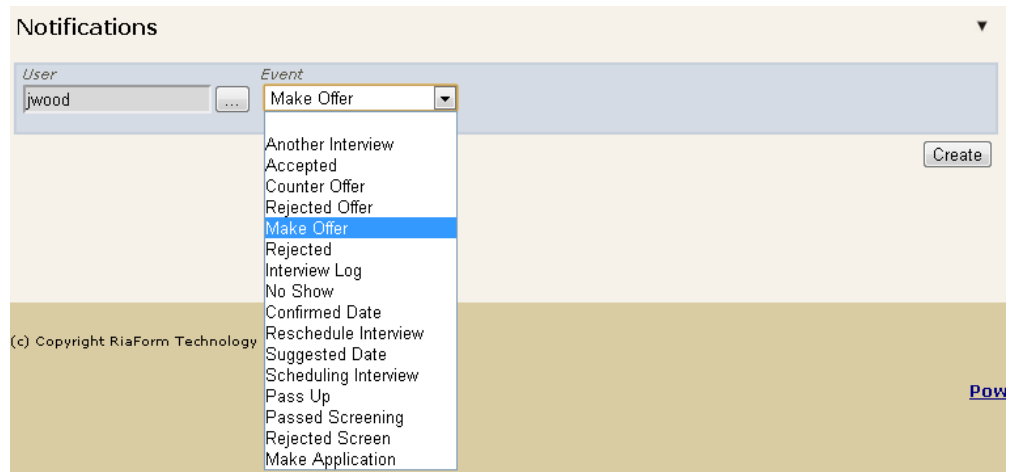


Figure 34: Notification entry, showing the workflow stages as events

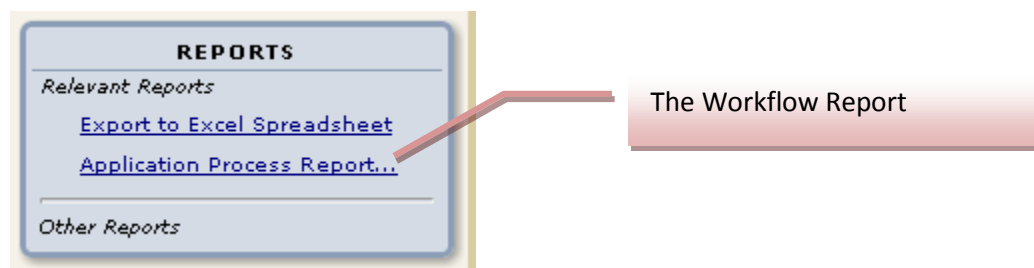
Note: You will need to define a user for each person subscribing to the workflow stages notifications. Additional licenses may be required to define those users.

7.3.3. Workflow Reports

7.3.3.1. Workflow Status Report

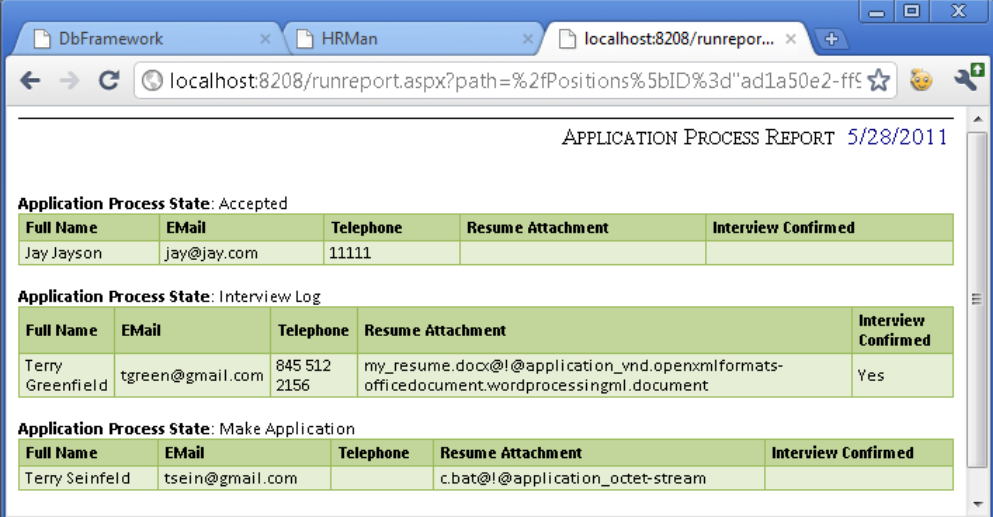
If you selected 'Create Report' when creating the workflow record, the system will automatically create a report for you that lists all of the outstanding tasks grouped by their workflow stage.

In the resultant application, the report will be available in the 'Reports' section on the right hand side of the screen. It will only be available when you are on the parent record of where the workflow is operating. If the workflow is operating on a top-level type, it will be available from anywhere.



The report will be named based on the name you gave to the workflow, followed by a 'Report' suffix.

When you go to run the report you will be given a filter form which you can optionally complete. The form lets you filter only the selected workflow stages in your report. If you just click create without selecting a stage, it will include all of them.



APPLICATION PROCESS REPORT 5/28/2011

Application Process State: Accepted

Full Name	EMail	Telephone	Resume Attachment	Interview Confirmed
Jay Jayson	jay@jay.com	11111		

Application Process State: Interview Log

Full Name	EMail	Telephone	Resume Attachment	Interview Confirmed
Terry Greenfield	tgreen@gmail.com	845 512 2156	my_resume.docx@!@application_vnd.openxmlformats-officedocument.wordprocessingml.document	Yes

Application Process State: Make Application

Full Name	EMail	Telephone	Resume Attachment	Interview Confirmed
Terry Seinfeld	tsein@gmail.com		c.bat@!@application_octet-stream	

The first seven (summary level) fields in the record will be included in the report, and each record will be grouped by their current stage.

This report can also be subscribed to, so that it is sent to your inbox each morning, for example. We'll discuss that in the next section.

7.3.3.2. *Tickler Reports*

If any field you create in your workflow stage has a data type of 'Date (Due Date)' the system will automatically create something called a 'tickler report'. This report will include all a list of all the records that are upcoming or just passed. You can subscribe to this report as an easy way to get a list of upcoming tasks, for example, or to track any tasks that have passed. For more information on tickler reports, see section 9.3.

7.3.4. *Workflow Report Subscriptions*

The workflow report is automatically set up for subscriptions. This allows any user to subscribe to have this report generated and emailed to them automatically at a specified frequency, and at a specified time.

Note: To subscribe to the report you will need a user record for your login. Defining user records may require additional licenses.

To enable the user or administrator to set up these subscriptions, a special area is added automatically in the parent of the record on which the workflow is operating. If the workflow is operating on a top-level record, the report subscriptions list will be maintained in the Configuration record, on the Configuration tab.

For more information about report subscriptions, please see section 9.5.

7.3.5. Custom Reports

Because the workflow operates on a record in the database defined with a regular data model, all of the features available to dbFramework and WorkflowFirst applications are also available to these records. One of the more powerful features is the reporting system.

You can create quick, ad-hoc reports using a simple point-and-click interface. These can include hierarchical sections, graphs and a variety of sophisticated layout structures. Alternatively you can utilize the custom reporting feature to design reports using either HTML or Microsoft Word, providing an unlimited degree of customization.

For more information on creating reports, see section 9.

7.3.6. Converging Workflow Paths

Sometimes you want the user to select all sub-stages before continuing, instead of just one. This is called converging paths.

To do this you will need to have each sub-stage set a field value (in the 'Fields to Collect' you specify a field, but place a value to set rather than asking the user). You will also have one sub-stage called a convergence point.

The convergent point will need to have a 'Condition' set that checks that all of those fields have been set before it will be displayed. This stage will remain invisible until each of those stages have been visited.

The other stages will have to be set to return to the parent stage every time they are executed. Also, to have them disappear when they are run, you should have a condition on that stage that checks the field hasn't been set.

7.3.7. Integrating Approval Workflow

A very common workflow situation is where a transaction must be approved by a certain group or individual. While you could model this using the workflow designer, WorkflowFirst and dbFramework make this even easier by supporting approval workflow as part of its security system.

This means that, as well as disallowing someone the rights to modify a certain area of the database, you can also require approval before that person can modify a certain part of the database. A specific user group signs up to govern changes in that area, and they will be emailed for approval before the change will be committed. All of this is handled automatically by the system, all you have to do is add the permissions.

Because workflow involves collecting information from the user and setting fields, these same permissions can be applied to fields in the record on which the workflow is operating. This allows you to easily combine approval workflow with your process workflow.

For example, in the above example of a process workflow, in the 'Make Offer' stage, you could set up a permission so that the offer amount field cannot be set to a value larger than \$100,000 without approval from the CFO. The amount and the user group requiring approval would both be configured in the application after it is published.

For more information on setting up approval workflow, see section 11.2.

7.3.8. External Initiators

Workflow is often a company-wide operation and therefore often involves more than one system. The ability to integrate data and triggers from external systems is critical where business process automation and efficiency is priority.

WorkflowFirst and dbFramework provide many ways to integrate third party systems.

The workflow stage is controlled by a field that is added when the application is published. The field is named based on the workflow name (with spaces removed) followed by 'State'. So, for example, if I named my workflow 'Ticket System', then the workflow state will be held in a field called 'TicketSystemState'.

The value of this field will be the name of the stage, with spaces removed.

This makes it very easy to initiate a workflow or to control the stage of a workflow from an external system. The system can simply utilize the XML interface described in section 21.6, for example, and update a record to set the state to a particular value. This would then send out any email notifications and start the workflow from that stage.

There are many other ways to interact with workflow, and these are covered in depth in section 21.

7.3.9. Mobile Access

As with any application created using dbFramework and WorkflowFirst, when the application is viewed on a mobile phone a special rendition of the application is provided that is simpler and suited to the size constraints of typical cellular phones.

7.3.10. Customizing Forms

One of the major benefits of using WorkflowFirst or dbFramework is that it creates forms for you, removing the necessity to spend time and resources with designing.

However sometimes you may want more control over how the forms are designed. WorkflowFirst and dbFramework provide several ways to customize the look and feel of your application, including providing HTML to replace the default form layout, providing HTML components for a specific field, or even re-skinning the entire application.

For more information on creating custom designs, refer to section 10.

7.4. Workflow Examples

7.4.1. Purchase Request

7.4.1.1. Requirements

The purchase request begins with an employee filling out a form specifying what they want to purchase, the approximate amount it will cost, the suggested vendor, and the project it is for. They will also provide details of how the product will be used, and how long the product will last before it will need to be upgraded or replaced.

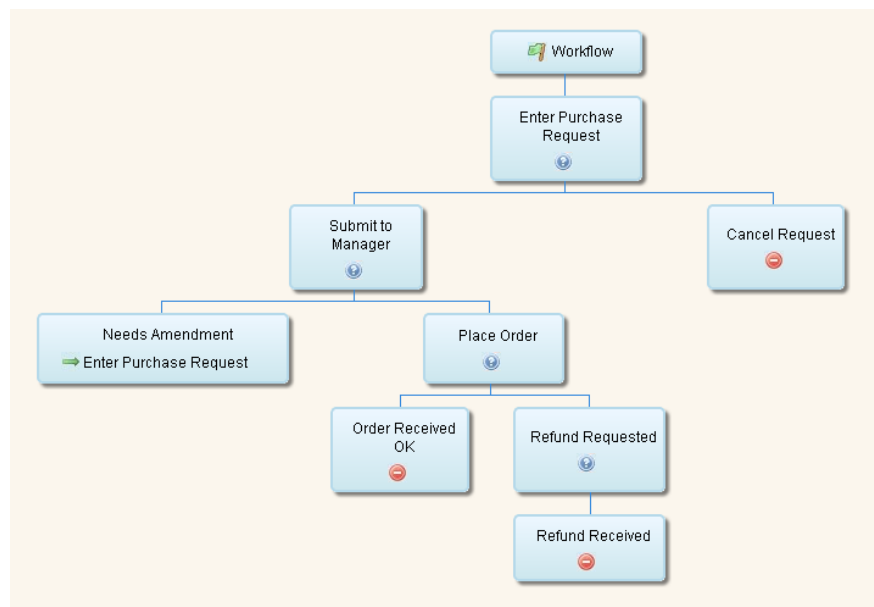
The request then goes to their manager who will approve the necessity of the product for the given project. They may accept it as-is, or they may reject it either

as being unnecessary, or that the chosen vendor is too expensive or otherwise unsuitable. At this point the originator will have to amend the request or give-up.

Once accepted, if the amount is greater than \$5,000 then the CFO needs to approve the purchase. It will then go to the accounting department who will place the order.

Once the order is received, the request will be closed and marked as fulfilled. Alternatively, if the order is never received then it will be marked as refunded.

7.4.1.2. Flowchart



7.4.1.3. Special Features

For the most part this is a simple flowchart. However the special case for the \$5,000 or over CFO approval will be implemented using the approval workflow feature, and so is not included in this flowchart. This works better because it means that the user(s) granting the approval, and the threshold amount, can both be defined at runtime without having to change the application itself.

7.4.2. Help Desk

7.4.2.1. Requirements

The Help Desk workflow starts with a customer having an issue they need to resolve. They provide information on the problem, such as the product they're

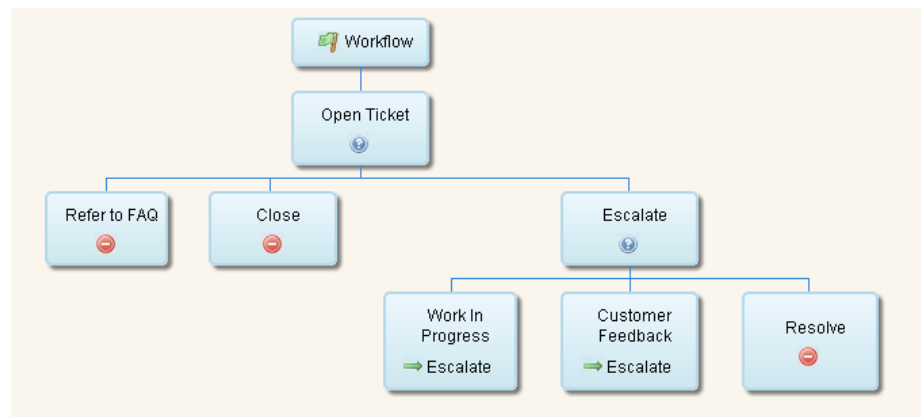
using, the version number, the steps to reproduce the problem, and also any identification of a support contract they have established. This is called a ticket.

The ticket is then moved to a primary support responder. They will screen the issue. If it is trivial, or can easily be found in a help file or FAQ, they will respond immediately and close the ticket. Otherwise they will escalate the ticket.

Once escalated, a technical expert will respond. They will have the option of asking the customer for more information, and the customer can respond. This can happen any number of times.

Finally the technical expert will mark the issue as resolved and it will be closed.

7.4.2.2. Flowchart



7.4.2.3. Special Features

Once escalated, the technical expert can effectively correspond with the customer using workflow. This consists of two workflow stages, 'Request Info' and 'Customer Feedback', both of which take the stage back to 'Escalate' at the end. This is effectively a loop. To break the loop the technical expert will select 'Resolved'.

7.4.3. Product Release

7.4.3.1. Requirements

At the end of product development, a number of stages are executed before the product can be released to the customers.

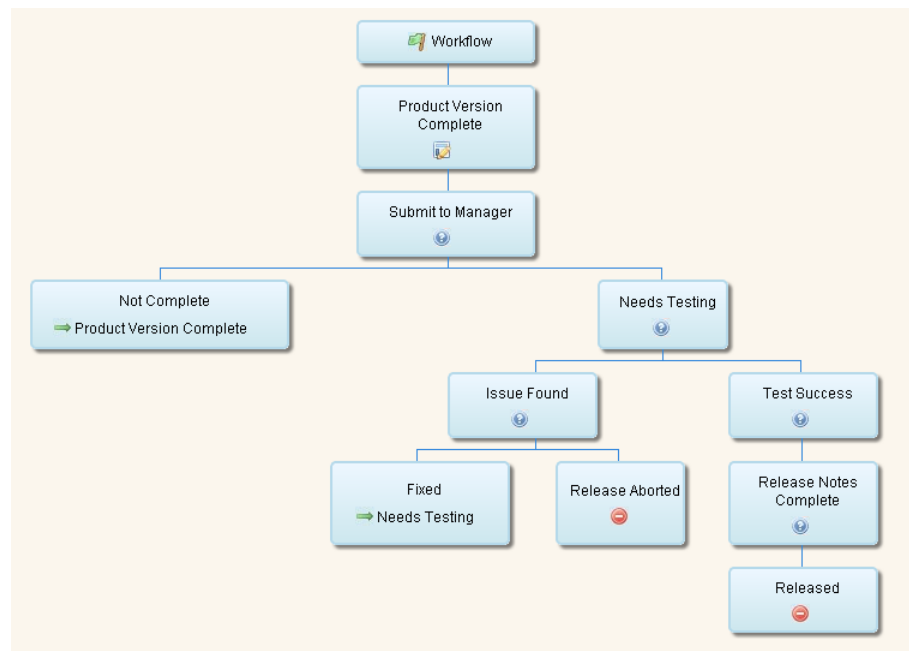
The first stage is for the manager to confirm that the necessary task list related to features in this release has been completed. The manager will enter those tasks into the workflow item so it can be passed on.

Once confirmed, a notification will go to Quality Assurance (QA) to start testing. They will be provided this with a deadline. They must test that all of the new features work as expected, and also that any existing functionality hasn't been broken. They will also have to confirm that the documentation is complete and meets the required standards.

If any problem is found, the work will return to development for completion, with a note explaining what was wrong.

Once QA are satisfied that the features are complete and the product is of an acceptable quality, a product release document will be generated. This will be sent to the marketing, sales, support and the deployment departments so they can confirm that they fully understand the new features and how to convey them to prospective, new or existing clients.

7.4.3.2. Flowchart



7.4.3.3. Special Features

If QA find a problem with the product, it is sent back to development. This effectively sends the stage back to the first stage, where it can be submitted to the manager and QA all over again. This can repeat any number of times. The loop is broken only when QA accept the product.

In the final stage, where the product release notes are submitted, we do not explicitly stage which departments must receive this. Rather this is set up through notification subscriptions. A representative from each department will subscribe to receive notification of when the product release is made.

7.4.4. Credit Application (Advanced)

7.4.4.1. Requirements

The credit application process starts with an entry by a potential borrower, the applicant, who wants to borrow some money.

The sales rep then receives that application and checks to make sure all of the required information s provided and that it calculates correctly. If not, it will be sent back to the applicant.

Once acceptable, the application is passed on to the credit rep. They perform their own validation and work with the applicant to ensure it is complete. They will then order a credit report. Once received they will determine the security requirement. They will then pass that requirement back to the applicant, who must respond with the necessary documentation.

The documentation will then be reviewed, and requests for more information or documentation will loop back to the applicant.

At this point the credit rep determines the credit approval. If declined, the sales rep and the applicant will be notified and the application closed.

If accepted, a formal notice will be sent to the applicant (through the word processing clerk). This will be signed by the credit rep before being sent to the applicant and to the sales rep. This must be signed by the applicant.

Once received back from the applicant, a copy will be sent to the loan operations department so that the necessary funds can be dispatched.

7.4.4.2. Flowchart

