

# Ferguson Moriyama

## Workflow for Umbraco

### Version 1.0

Copyright © 2011, Ferguson Moriyama Limited. All rights reserved

1	Introduction .....	4
2	Prerequisites .....	4
2.1	Requirements.....	4
2.2	Downloading .....	4
2.3	Evaluation .....	4
2.4	Purchasing a License .....	5
3	Installing.....	5
3.1	Uninstalling .....	5
4	What is workflow for Umbraco?.....	5
4.1	Screencasts .....	6
4.2	Workflow Configuration .....	6
4.3	Workflow Tasks.....	6
4.3.1	Decision Task.....	6
4.3.2	Delay Task .....	7
4.3.3	Runnable Task .....	7
4.4	Instantiation Criteria.....	7
4.5	Workflow Instances .....	8
5	Configuring Workflow for Umbraco.....	9
5.1	Creating a Workflow Configuration .....	9
5.2	Creating an Instantiation Criteria.....	11
5.3	Testing an Instantiation Criteria.....	12
6	End User Information.....	13
6.1	Starting a workflow .....	13
6.2	Checking and transitioning your Workflows.....	14
7	Developer Information.....	15
7.1	Creating Runnable Tasks.....	16
7.2	Creating Decision Tasks.....	18
7.3	Creating Delay Tasks .....	18
7.4	Globalizing Workflow for Umbraco .....	19
7.5	Logging .....	20
8	Support.....	21
9	Revision History .....	21

9.1 Version 1.0.3 ..... 21

10 License..... 21

## 1 Introduction

The simplest way to become familiar with Workflow for Umbraco is to watch our screencasts:

<http://vimeo.com/channels/253103>

Workflow for Umbraco allows you to author custom workflows using a drag and drop UI. You can have users manually push content, media etc. to workflows - or you can setup you workflows to instantiate upon API events.

Workflow attachments are CMS Nodes so workflows can run against documents, media, users, document types or any other Umbraco object.

Workflow GUI can be written for your language of choice without rebuilding - you simply provide an XML file for the appropriate .net culture.

Workflow tasks are POCOs which are decorated with a UI class so you can add your own tasks and expose properties to the GUI within minutes.

The project is built on Spring.net so core components such as storage and runtime can be customised and re-written without re-compiling.

Workflow state is persistent across app pool recycles and server reboots.

## 2 Prerequisites

### 2.1 Requirements

Workflow for Umbraco requires:

- Umbraco 4.7.1 - using SQL server
- .NET 4 – full trust

Previous versions of Umbraco and other databases are not supported.

### 2.2 Downloading

Workflow for Umbraco can be downloaded from the following URL:

<http://our.umbraco.org/projects/backoffice-extensions/workflow>

### 2.3 Evaluation

Workflow for Umbraco can be installed in an Umbraco instance using the domain name **localhost** or any domain ending in **.local**. Other domains will require a license.

## 2.4 Purchasing a License

You can buy a Workflow for Umbraco license from the Umbraco Deli.

Licenses are available on the following terms:

- Domain
- IP Address
- Unlimited

To purchase a license please visit <http://our.umbraco.org/projects/backoffice-extensions/workflow>

## 3 Installing

In the developer section of the Umbraco GUI:

- Expand the packages node of the navigation tree.
- Click on install local package.
- Click on the browse button in the right pane and locate your downloaded version of Workflow for Umbraco.
- Click on the Load Package button.
- Confirm any further prompts.

Installing via the Umbraco package repository is also supported.

### 3.1 Uninstalling

Uninstalling via the Umbraco package manager is not supported.

## 4 What is workflow for Umbraco?

This section is an overview of Workflow for Umbraco concepts only and does not contain any technical or implementation details.

## 4.1 Screencasts

The quickest way to get familiar with Workflow for Umbraco is to watch our screencasts:

<http://vimeo.com/channels/253103>

## 4.2 Workflow Configuration

A workflow configuration consists of one or more **workflow tasks**. A workflow configuration should have a **Start Task** – the task that is first run by the **workflow runtime** when it creates an **instance** of this workflow.

Workflow tasks have one or more outcomes or **transitions**. When a task completes it will choose a **transition** which will lead to another task or the end of the workflow.

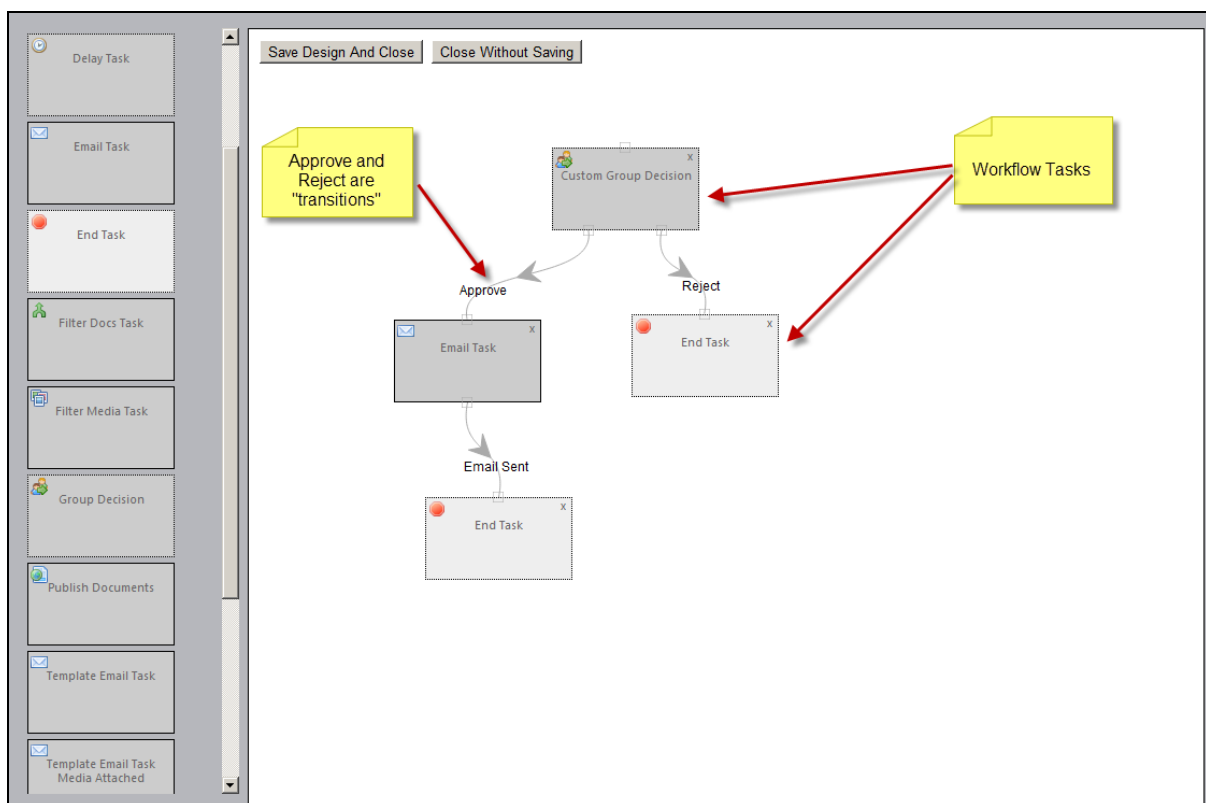


Figure 1 Workflow configuration in the designer

## 4.3 Workflow Tasks

There are 3 types of workflow task:

### 4.3.1 Decision Task

A **decision task** will stop the workflow and wait for the user to choose the next **transition**. Workflow for Umbraco ships with two out of the box decision tasks:

- Group Decision Task

- User Decision Task

A **Group decision task** allows users who are members of specific Umbraco groups to transition the workflow.

A **User decision task** allows a specific set of named Umbraco users to transition the workflow.

Developers may create their own decision tasks.

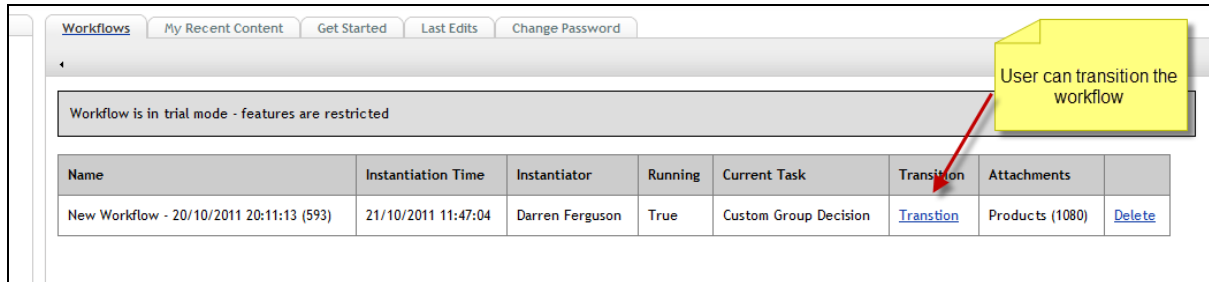


Figure 2 A decision task waiting for transition on the Umbraco dashboard

### 4.3.2 Delay Task

A **delay task** will cause the workflow to pause until a certain interval has passed. Workflow for Umbraco ships with two out of the box Delay tasks:

- Delay Task
- Wait Until Time

**Delay Task** waits for an interval which can be specified in minutes seconds or hours.

**Wait Until Time** waits until the next specified Hour and minute in a 24 hour period.

### 4.3.3 Runnable Task

A **runnable task** will execute some .NET code and inform the runtime of its transition afterwards.

## 4.4 Instantiation Criteria

**Instantiation Criteria** is a set of circumstances under which a workflow configuration will be started – or **instantiated**.

Instantiation Criteria consists of:

- Events – that cause instantiation
- User or User Types – that cause instantiation

Workflow may also be **manually instantiated** by a user if configuration allows.

**Edit Criteria**

Name  
New Instantiation Criteria

Active

Allow Manual Instantiation

Workflow Configuration  
New Workflow - 20/10/2011 20:11:13

Instantiating Events

- umbraco.cms.businesslogic.web.Domain.AfterDelete
- umbraco.cms.businesslogic.web.Domain.AfterSave
- umbraco.cms.businesslogic.web.Domain.BeforeDelete
- umbraco.cms.businesslogic.web.Domain.BeforeSave
- umbraco.cms.businesslogic.web.Domain.New
- umbraco.cms.businesslogic.web.StyleSheet.AfterDelete
- umbraco.cms.businesslogic.web.StyleSheet.AfterMove
- umbraco.cms.businesslogic.web.StyleSheet.AfterNew
- umbraco.cms.businesslogic.web.StyleSheet.AfterSave

Cancel Instantiating Event

Criteria Operand  
And

User Types

Figure 3 The Instantiation Criteria dialogue

## 4.5 Workflow Instances

**Workflow instances** are *running instances* of **Workflow configurations**.

When you install workflow for Umbraco the instances dashboard is installed in the content section of Umbraco.



Name	Instantiation Time	Instantiator	Running	Current Task	Transition	Attachments	
New Workflow - 20/10/2011 20:11:13 (593)	21/10/2011 11:47:04	Darren Ferguson	True	Custom Group Decision	<a href="#">Transition</a>	Products (1080)	<a href="#">Delete</a>

Figure 4 Workflow instances on the Umbraco dashboard

## 5 for Umbraco

## Configuring Workflow

Before reading this section it is recommended that you watch the introductory screencast which runs through all of the steps detailed below: <http://vimeo.com/30190458>

### 5.1 Creating a Workflow Configuration

- In the developer section of Umbraco choose the Workflow Configuration tab
- Click the **Create Workflow Configuration** button
- Your new configuration is displayed in the table

Description	Type	Active			
New Workflow - 20/10/2011 20:11:13	UmbracoWorkflowConfiguration	False	<a href="#">Delete</a>	<a href="#">Edit Properties</a>	<a href="#">Edit Design</a>

Figure 5 Creating a new workflow configuration

- Click the **Edit Properties** link
- In the resulting dialogue give your workflow configuration a descriptive name using the **Name** input field.

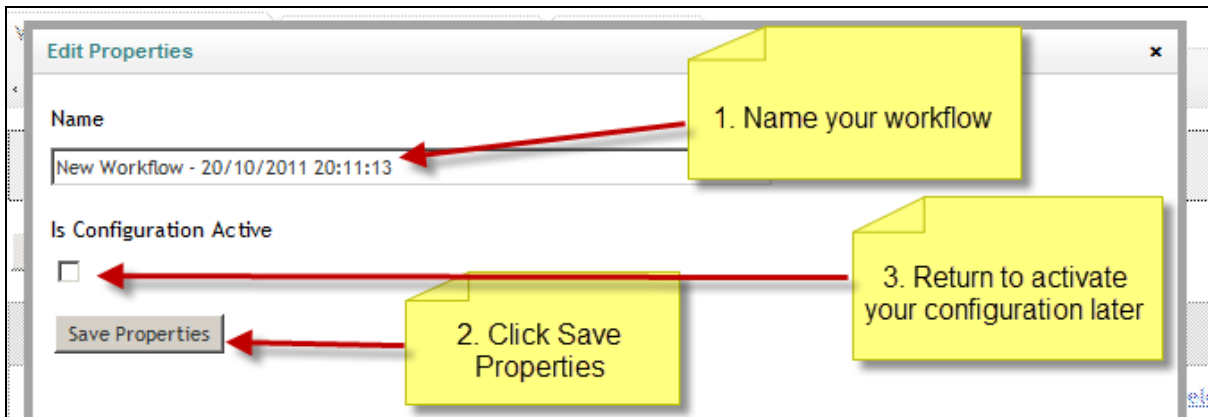


Figure 6 Setting basic workflow properties

- Click the **Save Properties** button
- When you finish designing your workflow you should return to this dialogue to make your configuration active.
- Click on **Edit Design** to launch the workflow designer

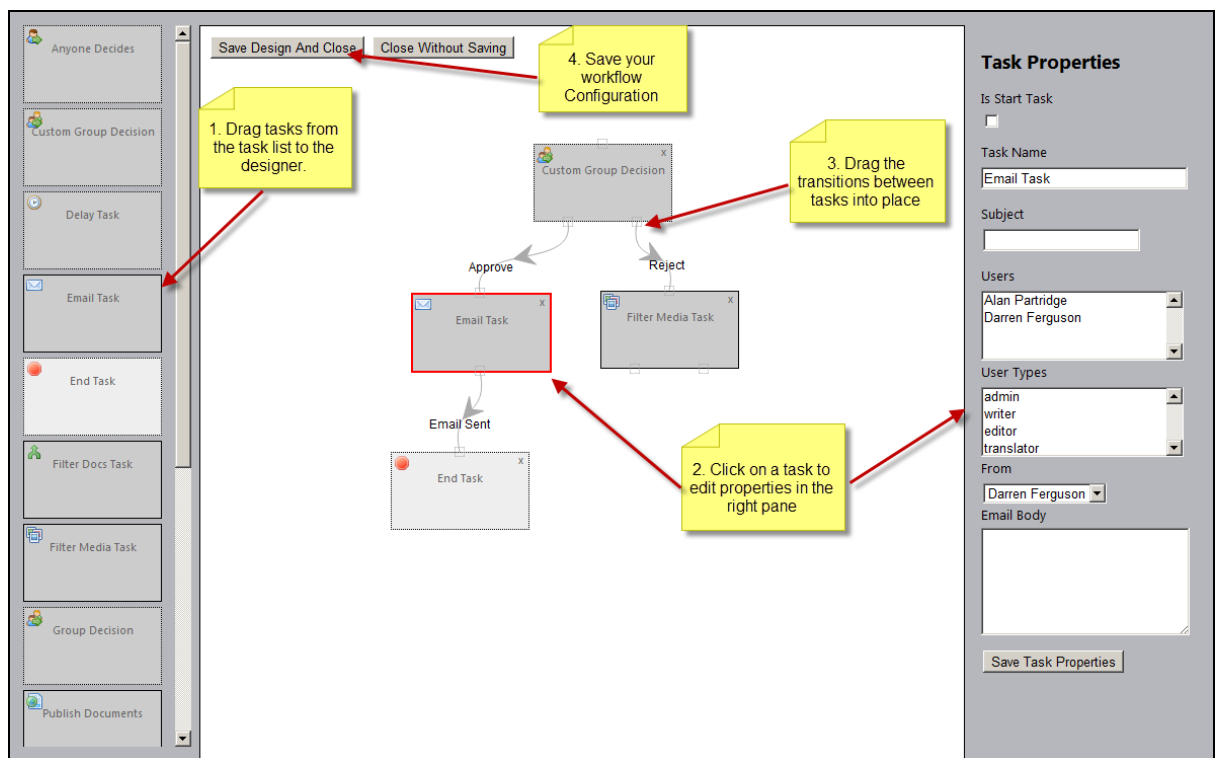


Figure 7 The workflow configuration designer

- Drag tasks from the **tasks list** (left hand pane) to the **designer** (central pane)
- To set the properties of a task click on it in the **designer** and edit properties in the **Task properties pane** (right pane). Be sure to click the **Save Task properties** button before navigating away.

- Drag your task transitions into place. Drag from the bottom of the **source task** to the top of the **destination task**. You can detach transitions by dragging the connection away from the **source task**.
  - Tasks may only be transitioned to by one other task
  - The designer does not currently report transitioning back to a task that has already been run (no looping)
- Click the **Save Design and Close** button to save your configuration
- Note: Your workflow design **must** contain **one** Start task otherwise the workflow runtime will throw an exception. You can make a task a start task by clicking on it in the designer and setting it as a Start Task in the **Task properties pane** (right pane).

## 5.2 Creating an Instantiation Criteria

- In the developer section of Umbraco choose the **Workflow Instantiation** tab
- Click the **Create new Criteria** button
- Click the **Edit Criteria** link in the criteria table corresponding to your new criteria

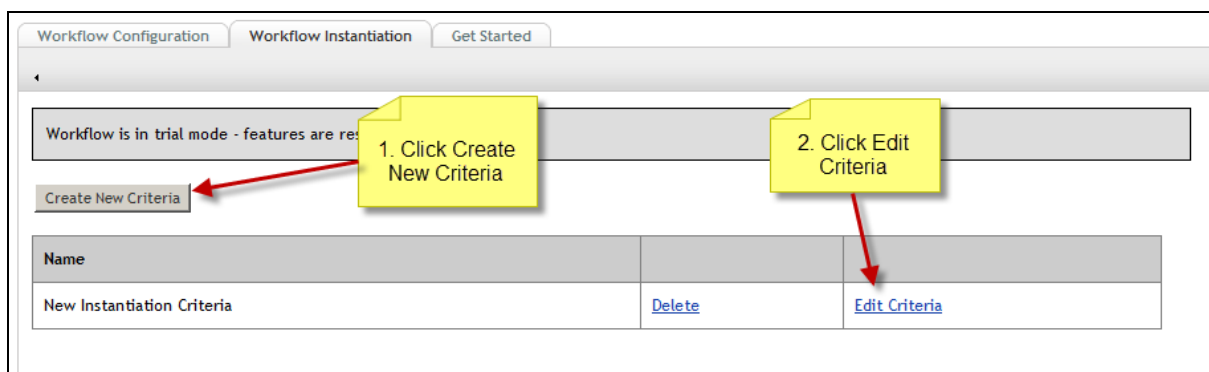


Figure 8 Creating a new workflow instantiation criteria

- Fill out the properties for your Criteria
  - Name – Descriptive Name
  - Active
  - Allow Manual Instantiation – Whether or not a user can send content to this criteria from their dashboard or whether it is event only

- Instantiating Events
- Cancel Instantiating Event – If the event has a cancel property then the runtime can set this cancel property on the instantiating event.
- Criteria Operand – Defines the logical operator to perform against further criteria
- User Types – Only apply criteria if current user is one of these types
- User – Only apply criteria if current user is one of these users

**Edit Criteria**

Name  
New Instantiation Criteria

Active

Allow Manual Instantiation

Workflow Configuration  
New Workflow - 20/10/2011 20:11:13

Instantiating Events

- umbraco.cms.businesslogic.web.Domain.AfterDelete
- umbraco.cms.businesslogic.web.Domain.AfterSave
- umbraco.cms.businesslogic.web.Domain.BeforeDelete
- umbraco.cms.businesslogic.web.Domain.BeforeSave
- umbraco.cms.businesslogic.web.Domain.New
- umbraco.cms.businesslogic.web.StyleSheet.AfterDelete
- umbraco.cms.businesslogic.web.StyleSheet.AfterMove
- umbraco.cms.businesslogic.web.StyleSheet.AfterNew
- umbraco.cms.businesslogic.web.StyleSheet.AfterSave

Cancel Instantiating Event

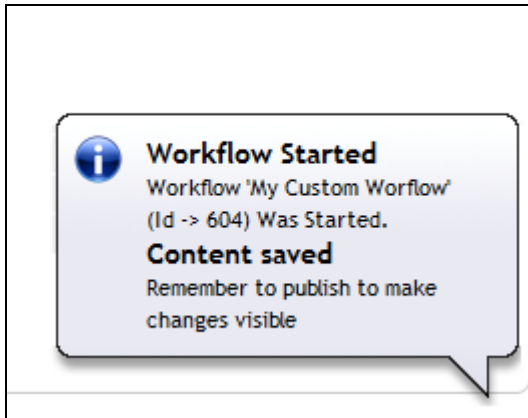
Criteria Operand  
And

User Types

Figure 9 The instantiation Criteria dialogue

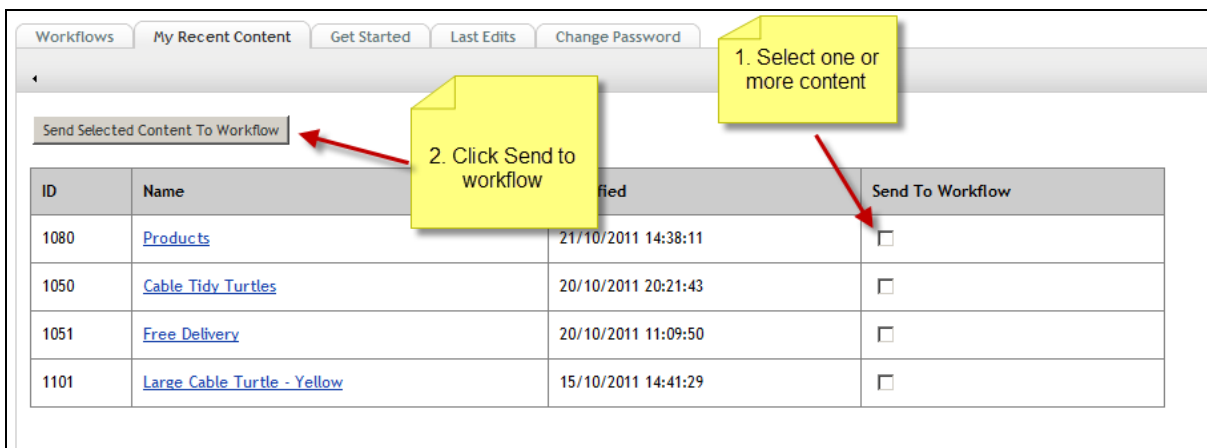
### 5.3 Testing an Instantiation Criteria

If an instantiation criteria is event based you should cause Umbraco to perform that event. If your criteria has caused a workflow to instantiate you will see a notification in the Umbraco speech bubble.



**Figure 10** The Umbraco speech bubble confirms a workflow instantiation

If you wish to verify a manual workflow instantiation go to the content section of Umbraco and choose the **My Recent Content** tab. From here you may send a recently modified document to your instantiation criteria.



**Figure 11** Manual workflow instantiation

*Note: If there is a current running workflow that was instantiated by the same event, user and content nodes then the workflow runtime will consider this instantiation to be a duplicate and cancel it.*

## 6 End User Information

### 6.1 Starting a workflow

To send content to a workflow:

- Go to the content section of Umbraco
- choose the **My Recent Content** tab
- Select one or more items and Click **Send to workflow**

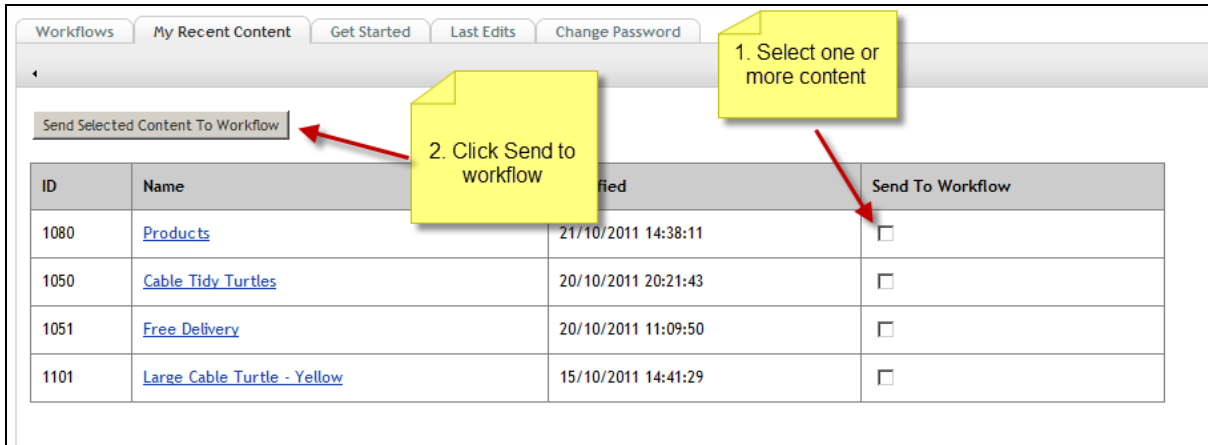


Figure 12 My recent content dialog

- In the resulting dialogue choose a workflow configuration
- Click **Send to Workflow**

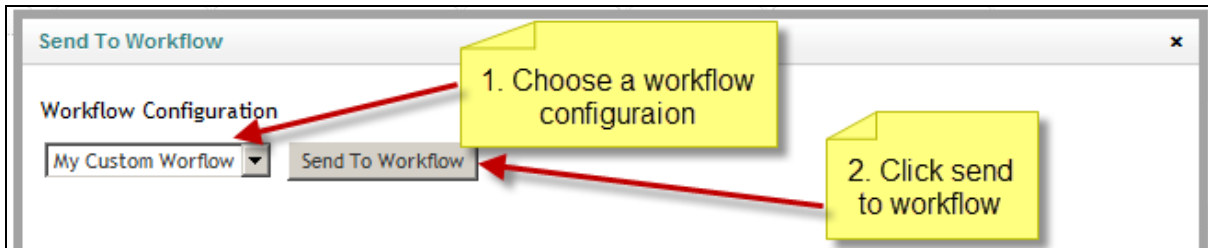


Figure 13 Sending an item to workflow

*Note: Your system administrator may have configured Umbraco so a workflow automatically starts when you perform common tasks such as saving and publishing content. In that case you'll be notified of the workflow starting via the Umbraco speech bubble.*

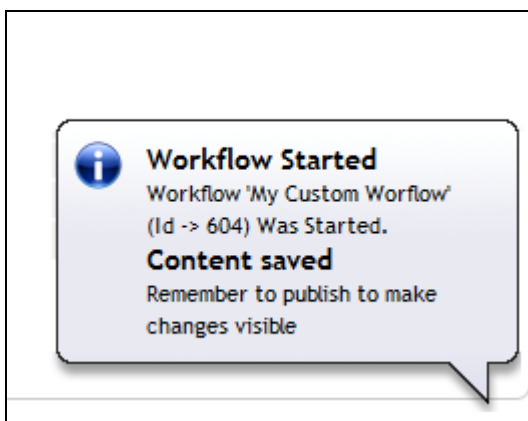


Figure 14 The Umbraco speech bubble shows a workflow has been started

## 6.2 Checking and transitioning your Workflows

In a typical scenario your system administrator will set up an email notification that will tell you when a workflow needs your attention.

- Goto the content section of Umbraco
- Choose the **Workflows** tab
- Any workflows that require your attention will have a **Transition** link in the transition column of the workflows table
- If you are happy with the content of the items attached to the workflow then click the transition link (note you will want to do this to reject items also).

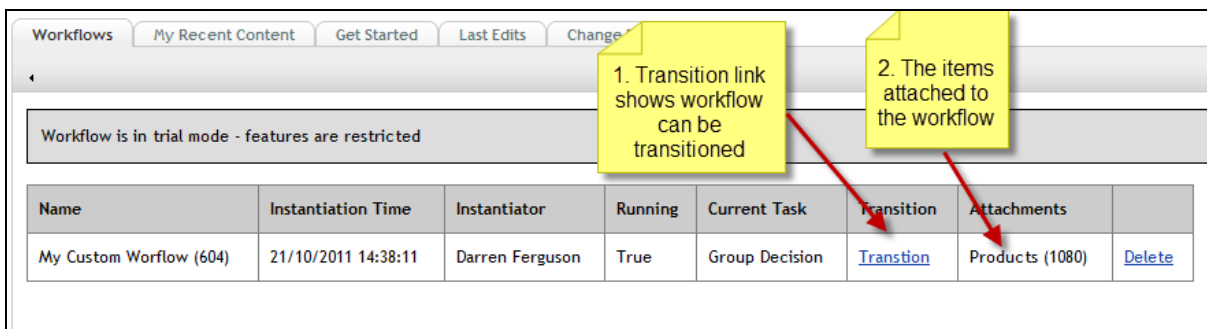


Figure 15 The transition link shows the workflow to be ready for transition by you

- In the resulting dialogue select your desired transition
- Click the **Transition** button

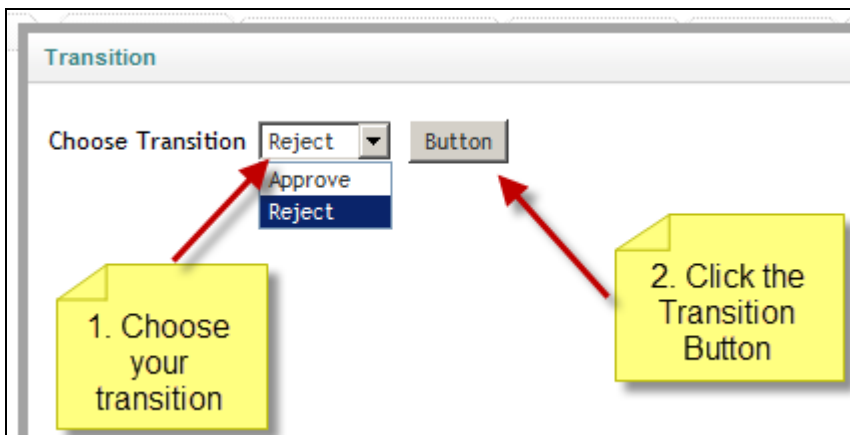


Figure 16 Choosing a transition

## 7 Developer Information

Workflow for Umbraco will discover any custom code you add by finding the appropriate implementations of its interfaces using reflection – this means that you just need to copy your .NET DLL files into the Umbraco /bin folder for your code to be discovered.

You may wish to consult the sample source code on bitbucket for examples of creating all of the task types detailed in this section: <https://bitbucket.org/darrenjferguson/workflow-samples>

## 7.1 Creating Runnable Tasks

- When creating a Runnable task you implement *IRunnableWorkflowTask*.
- Inheriting from Base Workflow task is the easiest way to get started.

```
1 [Serializable]
2 public class BitLyUrlShorteningTask : BaseWorkflowTask, IRunnableWorkflowTask
3 {
```

- Your class should tell the designer about its possible outcomes using the *AvailableTransitions* List
- Workflow tasks must always be marked as *Serializable*.

```
1 [Serializable]
2 public BitLyUrlShorteningTask() : base()
3 {
4     // A workflow task should let the designer know what possible outcomes
5     AvailableTransitions.Add("done");
6 }
```

- *IRunnableWorkflowTask* demands that you implement a *Run* method.
- For Umbraco workflows you can cast the workflowInstance parameter to an *UmbracoWorkflowInstance* to access Umbraco specific properties.

```
1 [Serializable]
2 // Implement the Run method of IRunnableWorkflowTask
3 public void Run(IWorkflowInstance workflowInstance, IWorkflowRuntime runtime)
4 {
5     // In Umbraco the workflowInstance should always be castable to an
6     UmbracoWorkflowInstance
7     var wf = (UmbracoWorkflowInstance) workflowInstance;
8     ....
9 }
```

- Your run method is responsible for informing the workflow runtime of the task outcome.

```
1 runtime.Transition(workflowInstance, this, "done");
```

- In Order that the designer knows how render the task you must create an *IWorkflowTaskEntityUi*

```
// An Entity Ui describes how the workflow task is rendered in the workflow
Designer
1 public class BitLyUrlShorteningTaskEntityUi : BaseWorkflowTaskEntityUi,
2 IWorkflowTaskEntityUi, IGlobalisable
3 {
4     ...
5 }
```

- Use the entity UI to describe task transitions
- To add CSS classes to your task in the designer



- **To add Controls to the workflow designer to set the public properties of the task.**

```

// An Entity Ui describes how the workflow task is rendered in the workflow
Designer
1 public BitLyUrlShorteningTaskEntityUi()
2     : base()
3 {
4     // Describe the workflow transition
5     TransitionDescriptions.Add("done", "URL Was shortened");
6
7     // Decorate your task with a custom CSS class in the designer.
8     // The workflow designer scans ~/umbraco/plugins/fmworkflow/css for custom
9     CSS files and includes all of them.
10    UiAttributes.Add("class", "bitlyTask");
11
12    // These properties explain how to present a Ui to get values for the
13    public properties of the workflow task.
14    UiProperties.Add(new BitLyApiKeyPropertyUi());
15    UiProperties.Add(new BitLyLoginPropertyUi());
16    UiProperties.Add(new DocumentTypePropertyUi());
17 }

```

- **The Entity UI should also implement a method to tell the designer the name of the workflow task**
- **And also a method that tells the designer which workflow task it is for**

```

1 // When passed an object returns a bool indicating whether this task can supply
2 // a UI for it.
3 public override bool SupportsType(object o)
4 {
5     return o.GetType() == typeof(BitLyUrlShorteningTask);
6 }
7 // Name of task
8 public override string EntityName
9 {
10    get { return "Bit.ly URL Shorten"; }
11 }

```

- **The UiProperties list of the Entity Ui is a collection of *IWorkflowUiProperty***
- **A Property UI describes how to get and set a public property of a workflow task using the workflow designer**

```

1 public class BitLyApiKeyPropertyUi : PropertyUi, IWorkflowUiProperty,
2   IGlobalisable
3 {
4     ....
5 }

```

- **The Property UI defines a Render Control**

```

1 public Control RenderControl { get; private set; }
2
3 public BitLyApiKeyPropertyUi()
4 {
5     // Just says that a TextBox will be used to get the property value from the
6     user.
7     RenderControl = new TextBox { ID = PropertyName, CssClass

```

```
= "workflowTextBox" };
}
```

- A getter and a setter for its value

```
1 // The workflow designer uses this getter and setter to give or take the value
  of the property.
2 public object Value
3 {
4     get { return ((TextBox)RenderControl).Text; }
5     set { ((TextBox)RenderControl).Text = (string)value; }
6 }
```

- And lastly implements two properties which return the name of the property to set and the description of the property

```
1 // Label for control when rendered in Ui
2 public string Label
3 {
4     get { return "Bit.ly API key"; }
5 }
6 // The workflow designer uses this getter and setter to give or take the value
7 of the property.
8 public object Value
9 {
10    get { return ((TextBox)RenderControl).Text; }
11    set { ((TextBox)RenderControl).Text = (string)value; }
12 }
```

## 7.2 Creating Decision Tasks

Our screencast on creating Decision tasks is at: <http://vimeo.com/30916953>

- The user decision task allows a list of specified Umbraco users to transition a task
- The group decision task allows a list of specified Umbraco user types to transition a task
- To create your own decision tasks implement *IDecisionTask*
- The workflow runtime will ask if a user can transition a task by calling the *CanTransition* method of *IDecisionTask*
- If you want to write your own GUI (aspx) to perform the transition override the *TransitionUrl* property of *IDecisionTask* a sample is provided in the code accompanying the screencast.

## 7.3 Creating Delay Tasks

For an overview of creating delay tasks you can watch the screencast at:

<http://vimeo.com/30873482>

You can download example source code from: <http://our.umbraco.org/FileDownload?id=3197>

- A delay tasks causes the workflow to pause.
- To implement your own you just implement *IDelayTask*
- The runtime will tell your task what time it was started by setting the *StartTime* property.
- The runtime will periodically ask your delay task whether it is complete by calling the *IsComplete* Method.
- You can use the *StartTime* property to perform time based calculations or collect user properties as with runnable tasks.
- Delay tasks can be good for delayed publish, tweet etc. and allow you to specify when subsequent tasks should happen.

## 7.4 Globalizing Workflow for Umbraco

- When the workflow runtime instantiates a task it will check whether the task implements *IGlobalisable*
- If it does then it will pass an instance of *IGlobalisationService* to the task using the *TheGlobalisationService* property
- The task can then call the *GetString* method to retrieve UI text
- When adding Ui Properties to the Entity UI you can use the *CreateGlobalisedObject* helper to pass a *GlobalisationService* to your Ui Properties.
- This is done pre-constructor to enable you to use the *GlobalisationService* in constructor logic.
- You can replace the default application *IGlobalisationService* implementation should you wish.

To set up Workflow UI languages:

- Take a copy of `~/config/fmworkflow/workflow.en-GB.spring.config`
- Rename it to match the Culture you are creating a UI for e.g. `workflow.da-DK.spring.config`
- In your file add your language text and ensure that the object name has a unique ID.
- Load your file in `workflow.spring.config`
- Register your culture in `workflow.globalisation.spring`

- Recycle your App pool

## 7.5 Logging

By Inheriting **BaseWorkflowTask** and referencing **log4net.dll** you will have a local log4net logger available called **Log** which you can use to log to the Workflow for Umbraco logs in your **~/App\_Data** folder.

## 8 Support

Ferguson Moriyama provides support for Workflow for Umbraco via our.umbraco support forums at <http://our.umbraco.org/projects/backoffice-extensions/workflow>

## 9 Revision History

### 9.1 Version 1.0.3

Released 24-11-2011

- Fixed issues with template email tasks.
- Added improved installer to allow logging and upgrades
- Added options to mail Instantiator and Node owner from base email tasks.

## 10 License

### UMBRACO DELI PACKAGE LICENSE AGREEMENT

This UMBRACO DELI PACKAGE is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. This UMBRACO DELI PACKAGE is licensed, not sold.

#### Definitions

UMBRACO DELI PACKAGE refers to the software products provided by the Umbraco Deli Vendor and not directly from Umbraco I/S. The Umbraco Deli Vendor information can be found by referencing the Umbraco Deli Package information.

#### End User License Agreement

This End User License Agreement ("EULA") is a legal agreement between you (either an individual or a single entity) and the Umbraco Deli Vendor and not Umbraco I/S with regard to the copyrighted Software (herein referred to as "UMBRACO DELI PACKAGE" or "SOFTWARE") provided with this EULA. The UMBRACO DELI PACKAGE includes computer software, the associated media, any printed materials, and any "online" or electronic documentation. Use of any software and related documentation ("Software") provided to you by the Umbraco Deli Vendor and not Umbraco I/S in whatever form or media, will constitute your acceptance of these terms, unless separate terms are provided by

the software supplier, in which case certain additional or different terms may apply. If you do not agree with the terms of this EULA, do not download, install, copy or use the Software. By installing, copying or otherwise using the UMBRACO DELI PACKAGE, you agree to be bound by the terms of this EULA. If you do not agree to the terms of this EULA, the Umbraco Deli Vendor is unwilling to license the UMBRACO DELI PACKAGE to you.

1. Eligible Licensees. This Software is available for license solely to SOFTWARE owners, with no right of duplication or further distribution, licensing, or sub-licensing. **IF YOU DO NOT OWN THE SOFTWARE, THEN DO NOT DOWNLOAD, INSTALL, COPY OR USE THE SOFTWARE.**

2. License Grant. The Umbraco Deli Vendor and not Umbraco I/S grants to you a personal, non-transferable and non-exclusive right to use the copy of the Software provided with this EULA. You agree you will not copy the Software except as necessary to use it on a single Umbraco site. You agree that you may not copy the written materials accompanying the Software. Modifying, translating, renting, copying, transferring or assigning all or part of the Software, or any rights granted hereunder, to any other persons and removing any proprietary notices, labels or marks from the Software is strictly prohibited. Furthermore, you hereby agree not to create derivative works based on the Software. You may not transfer this Software.

3. Copyright. The Software is licensed, not sold. You acknowledge that no title to the intellectual property in the Software is transferred to you. You further acknowledge that title and full ownership rights to the Software will remain the exclusive property of the Umbraco Deli Vendor and you will not acquire any rights to the Software, except as expressly set forth above. All copies of the Software will contain the same proprietary notices as contained in or on the Software. All title and copyrights in and to the UMBRACO DELI PACKAGE, the accompanying printed materials, and any copies of the UMBRACO DELI PACKAGE, are owned by the Umbraco Deli Vendor and not Umbraco I/S. The UMBRACO DELI PACKAGE is protected by copyright laws and international treaty provisions. You may not copy the printed materials accompanying the UMBRACO DELI PACKAGE.

4. Reverse Engineering. You agree that you will not attempt, and if you are a corporation, you will use your best efforts to prevent your employees and contractors from attempting to reverse compile, modify, translate or disassemble the Software in whole or in part. Any failure to comply with the above or any other terms and conditions contained herein will result in the automatic termination of this license and the reversion of the rights granted hereunder to the Umbraco Deli Vendor.

5. Disclaimer of Warranty. The Software is provided "AS IS" without warranty of any kind. The Umbraco Deli Vendor and Umbraco I/S disclaims and makes no express or implied warranties and specifically disclaim the warranties of merchantability, fitness for a particular purpose and non-infringement of third-party rights. The entire risk as to the quality and performance of the Software is with you. The Umbraco Deli Vendor does not warrant that

the functions contained in the Software will meet your requirements or that the operation of the Software will be uninterrupted or error-free. The Umbraco Deli Vendor and not Umbraco I/S IS NOT OBLIGATED TO PROVIDE ANY UPDATES TO THE SOFTWARE.

6. Limitation of Liability. The Umbraco Deli Vendor's entire liability and your exclusive remedy under this EULA shall not exceed the price paid for the Software, if any. In no event shall the Umbraco Deli Vendor or Umbraco I/S be liable to you for any consequential, special, incidental or indirect damages of any kind arising out of the use or inability to use the software, even if the Umbraco Deli Vendor and Umbraco I/S or its supplier has been advised of the possibility of such damages, or any claim by a third party.

7. Rental. You may not loan, rent, or lease the SOFTWARE.

8. Upgrades. If the SOFTWARE is an upgrade from an earlier release or previously released version, you now may use that upgraded product only in accordance with this EULA. If the UMBRACO DELI PACKAGE is an upgrade of a software program which you licensed as a single product, the UMBRACO DELI PACKAGE may be used only as part of that single product package and may not be separated for use on more than Umbraco site.

9. Product Support. Product support for the UMBRACO DELI PACKAGE IS provided by the Umbraco Deli Vendor and not Umbraco I/S. For product support, please contact the Umbraco Deli Vendor using the appropriate online support request forms.

10. No Liability for Consequential Damages. In no event shall the Umbraco Deli Vendor or Umbraco I/S be liable for any damages whatsoever (including, without limitation, incidental, direct, indirect special and consequential damages, damages for loss of business profits, business interruption, loss of business information, or other pecuniary loss) arising out of the use or inability to use this "Your Company" product, even if the Umbraco Deli Vendor and Umbraco I/S has been advised of the possibility of such damages. Because some states/countries do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

11. Indemnification By You. If you distribute the Software in violation of this Agreement, you agree to indemnify, hold harmless and defend the Umbraco Deli Vendor and Umbraco I/S from and against any claims or lawsuits, including attorney's fees that arise or result from the use or distribution of the Software in violation of this Agreement.

Umbraco I/S

Lindholm havnevej 31

DK-5800 Nyborg

Denmark

<http://umbraco.com/>