Haumohio

Haumohio RSS Developer's Pack

Haumohio RSS Developer's Pack

Printed: April 2006 in Rotorua, New Zealand

Table of Contents

1 Introducing the RSS Developer's Pack

 The RSS Developer's pack contains controls and components for Microsoft .Net developers that easily add RSS News Feed capability to products under development. The components in this pack are designed to help you as a developer to quickly and easily harness the power of RSS to integrate your company's news and announcements into your products, without having to implement all the nitty-gritty parts of syndicated news feeds.

Take the hassle out of remote news feeds

The Haumohio RSS components quickly look up a news feed formatted XML file on your website, which contains your current news. Changing the XML file, changes what *all* your users see in one easy stroke. The feed could contain product update announcements, answers to the latest burning FAQ, event announcements, or anything that you desire to communicate to the users of your software. Haumohio FeedWrite could be used to create the XML file in the correct shape, and with all of the required fields.

The components in this pack are able to read from a number of news feed standards, including:

- RSS v0.9
- RSS 1.x
- RSS 2.x
- ATOM 0.9
- RDF

The components in this pack integrate into Microsoft's Visual Studio; simply add them to a form, set the parameters of the news feed, and your done!

2 What's New in v3

Version 3 of the developers pack is entirely based on version 2.0 of the .NET framework.

It takes advantage of new framework features such as the tool strip with embedded pictures, and generic collections to bring you a richer and more stable environment.

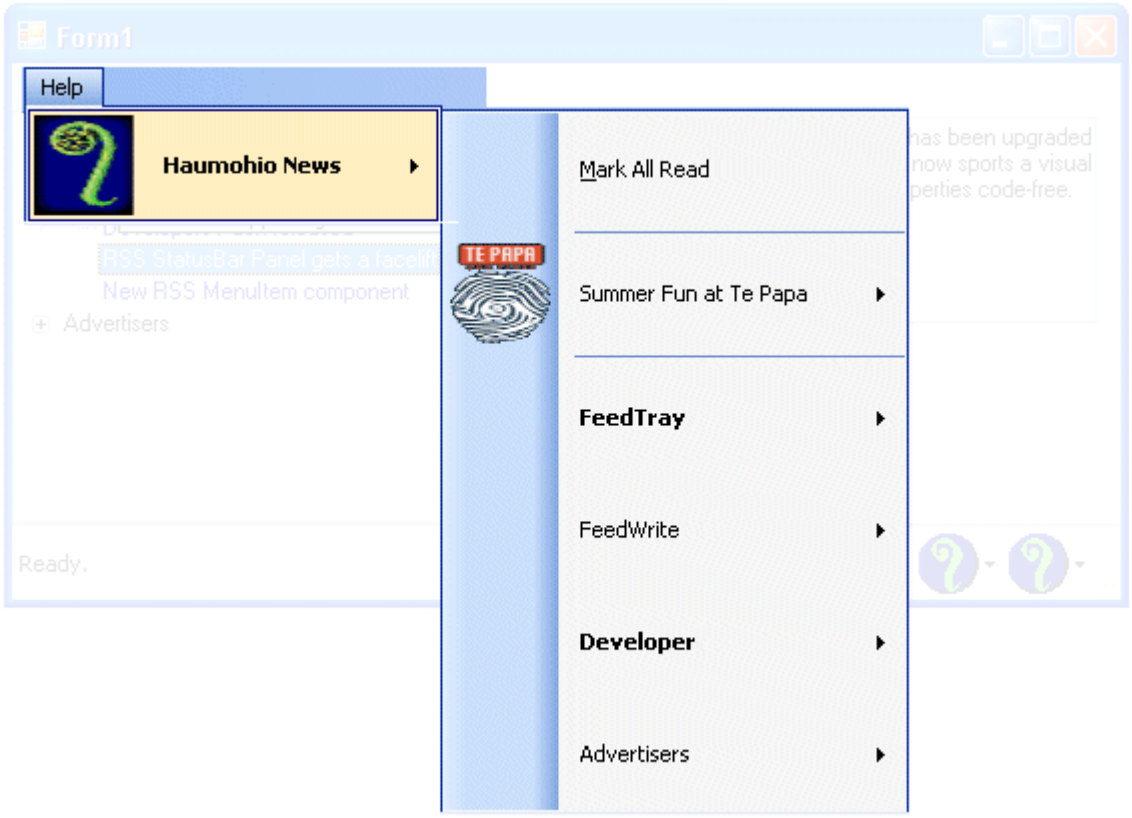
Version 3.1 now reads feeds in the RDF format, and also supports the WFW commenting system. This means that if the feed contains the tags wfw:comment and / or wfw:commentRss, the user can make and read comments on any article - right in the menu!

3 Components in this pack

3.1 The RSS Menu Item Component

3.1.1 Introducing the RSS Menu Item

The RSS Menu appears to a user to be just like any other tool strip menu item, but *SURPRISE!* its contents change daily. How do they do that?



The RSS Menu Item can be added to any menu just like any other standard ToolstripMenuItem object. When the command is given it quietly goes off to read its designated news feed (in a separate thread - of course) and populates its sub-menu items with the up-to-the-second news articles. Any number of RSS MenuItems can exist in a project, form, or even menu, without clashing.

The control loads its data from the configured internet location in a separate thread so as not to interrupt the flow of the interface. If an internet connection or the news feed is not available, then it will indicate as such and stop using up resources in fruitless searching. This means the user is not disadvantaged in using your product if not connected.

Keeping track of what has been read or not

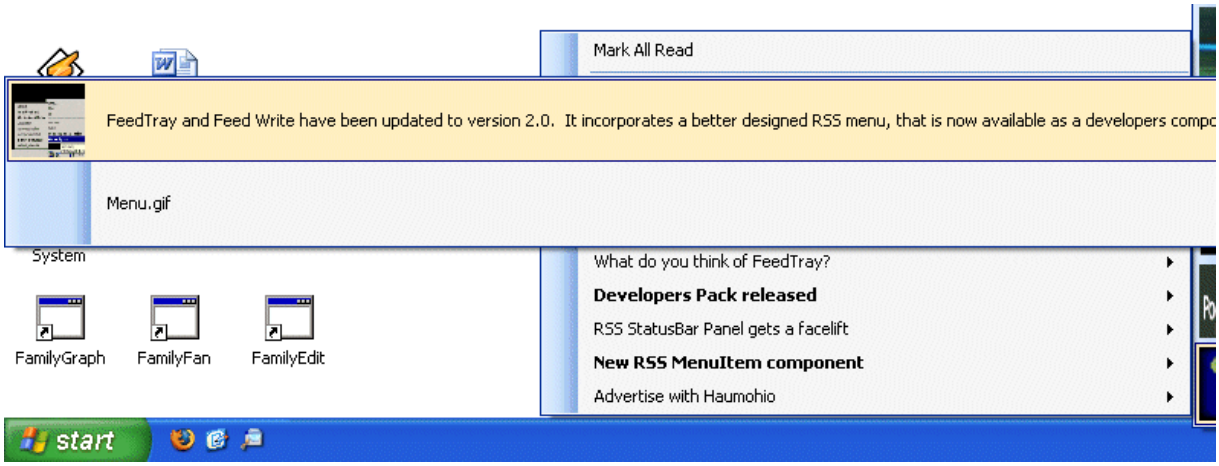
It even keeps track of which articles the current user has read, and highlights the unread articles by rendering them as in bold font. This means the user only has to open the containing menu (e.g. the Help menu in this case), and see if the news feed is rendered in bold, in order to see if there is anything new. If the articles are categorised, the unread highlighting is propogated up to the

containing category. So, in the example above both "Haumohio News" and "FeedTray" are highlighted, because the article entitled "Version 1.02 of FeedTray" has not yet been read by the current user. Optionally, a number in brackets indicating the nunber of unread articles may be shown also.

This is implemented by saving the unread article titles in a xml file in the user's application data folder. If, on the next read of the news feed, an article has been removed from the feed, then it is removed from the list as well. Therefore, there will only be a maximum of the number of entries in the file, as there is articles in the current news feed.

What happens when an article item is clicked

When an article in the news feed is selected by the user a short description of what the article is about (i.e. the content of the news feed file) in shown as below, along with a thumbnail of an enclosed image, if the image is not too big.

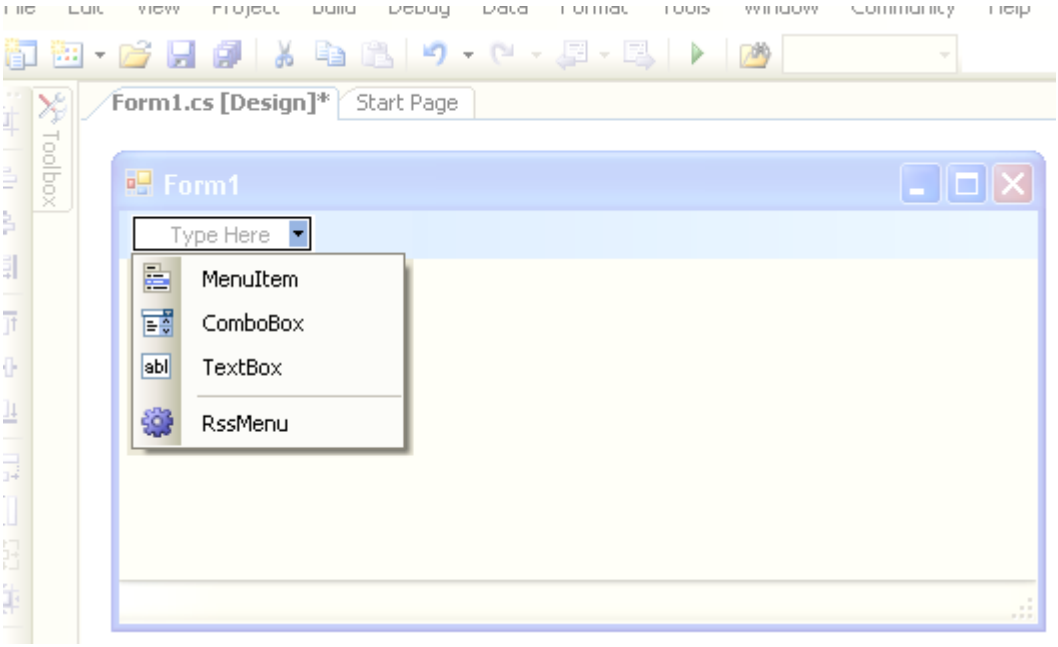


Cicking on the text or the logo of the article description closes the menu and launches the default web browser, opening the link nominated within the article. This could be a full article in an online magazine, a video / music file, or an installer for your next upgrade - basically anything you can point a URL at. Haumohio FeedWrite could be used to create the news feed file in the correct shape, and with all of the required fields. Any enclosed media (including the article's logo) is listed as another menu item below the main description item. Clicking the enclosed media's title launches the URL to the media in a similar fashion.

3.1.2 Configuring RSS MenuItem

Configuring the RSS MenuItem is as simple as telling it where the news feed is, your licence details, and where the menu should be displayed.

Once a reference to the Developer's Pack had been added to the project, adding an RssMenu can be added to any MenuStrip (main or context) by simply selecting it from the dropdown list in the designer.



Configuration of the RssMenu is performed through a few properties on the component, that may be entered using the visual designer in Microsoft Visual Studio, or just as easily with a text editor. These properties are:

URL	The URL of the news feed file itself; usually on your company's website
CompanyRegistrationEmail	Your registered email, used as a key to authorise the URL as registered. See Licensing components.
AutoLoad	Automatically begin loading the news feed once the application is up and running
Categorise	A flag indicating whether the feed should be split into sub-menus based on each articles "category" attribute. Note that this is an optional RSS attribute and therefore may not exist on the chosen news feed

Activating the component

The actual loading of the data from the news feed is entirely under your control. There is a single command given to the control - namely **Fetch()**. This command can be given at any time, or even multiple times, during the lifetime of the application. For instance, Fetch() can be called within the *Load* event of the containing form so that the components are ready when the form is shown, or delayed until later if the *Load* event is a "busy" time. Alternately, the **AutoLoad** property may be set at design time, and the component will begin downloading when *EndInit()* is called in the designer code.

Configuring without an IDE

The RSS MenuItem can be configured using the following code snippet:

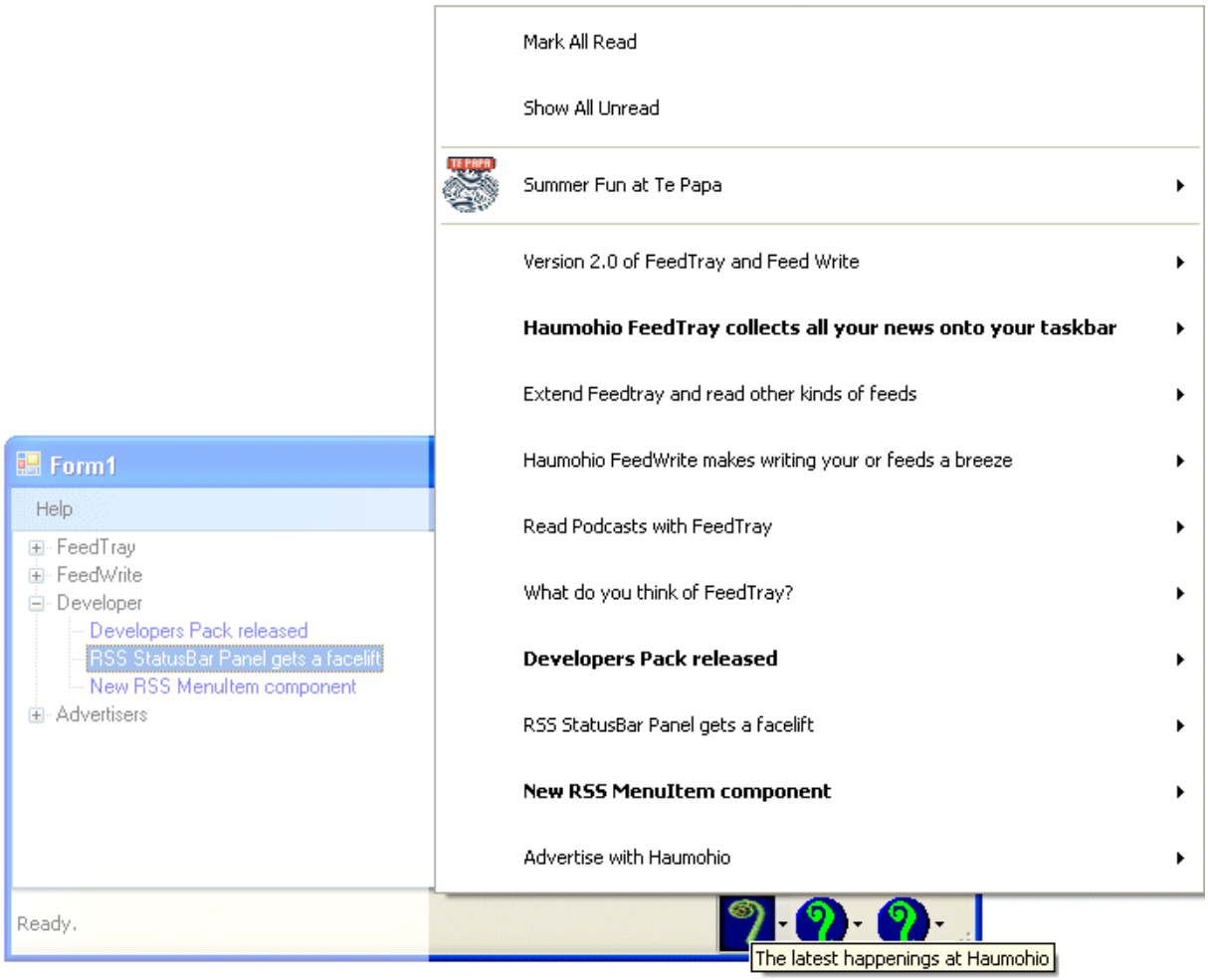
```
[C#]
using Haumohio.RSS;

       private RssMenu rssMenu1 = new RssMenu();
```

```
...
       ((System.ComponentModel.ISupportInitialize)(this.rssMenu1)).BeginInit();

      rssMenu1.Categorise = true;
      rssMenu1.CompanyRegistrationEmail = "sales@haumohio.com";
      rssMenu1.RssFeedUrl = "http://www.haumohio.com/haumohio.xml";

       this.helpToolStripMenuItem.DropDownItems.Add( this.rssMenu1 );

       ((System.ComponentModel.ISupportInitialize)(this.rssMenu1)).EndInit();
...
      rssMenu1.Fetch();

[VB]
Imports Haumohio.RSS

      Dim rssMenu1 AS RssMenu = New RssMenu()
...
      rssMenu1.BeginInit

      rssMenu1.Categorise = True
      rssMenu1.CompanyRegistrationEmail = "sales@haumohio.com"
      rssMenu1.RssFeedUrl = "http://www.haumohio.com/haumohio.xml"

       helpToolStripMenuItem.DropDownItems.Add( this.rssMenu1 )

      rssMenu1.EndInit
...
      rssMenu1 .Fetch()
```

3.2 The RSS DropDownButton Component

3.2.1 Introducing The RSS DropDownButton

The RSS DropDownButton looks like an unobtrusive icon in the status bar normally found at the bottom of nearly every Windows product. It encapsulates an RSS MenuItem and brings the news feed more to the forefront of the user's attention.

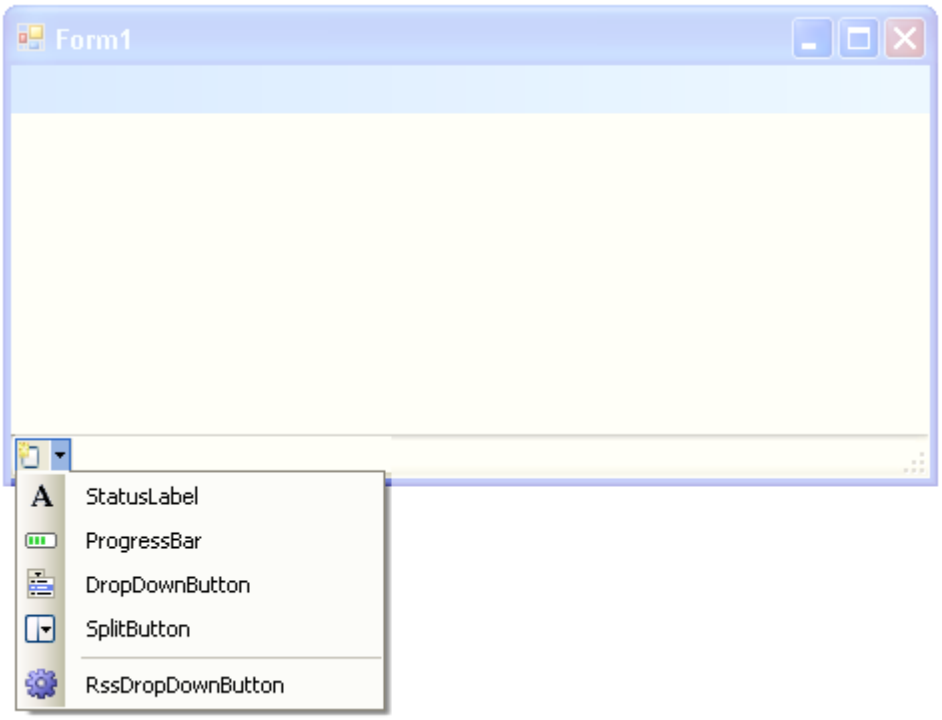


As soon as the newsfeed has been read the icon changes to a thumbnail of the image nominated by the newsfeed as its image. Also, whenever a new (unread) article is discovered the icon flashes with the number of currently unread articles. This means the user immediately knows within a few seconds of starting the application that there is new news to be read.

3.2.2 Configuring RSS DropDownButton

Configuring the RSS DropDownButton is as simple as telling it where the news feed is, your licence details, and where the menu should be displayed.

Once a reference to the Developer's Pack had been added to the project, adding an RssDropDown can be added to any ToolStrip or StatusStrip by simply selecting it from the dropdown list in the designer.



Configuration of the RssDropDown is performed through a few properties on the component, that may be entered using the visual designer in Microsoft Visual Studio, or just as easily with a text editor. These properties are:

URL	The URL of the news feed file itself; usually on your company's website
CompanyRegistrationEmail	Your registered email, used as a key to authorise the URL as registered. See Licensing components.
AutoLoad	Automatically begin loading the news feed once the application is up and running
Categorise	A flag indicating whether the feed should be split into sub-menus based on each articles "category" attribute. Note that this is an optional RSS attribute and therefore may not exist on the chosen news feed

Configuring without an IDE

The RssDropDown can be configured using the following code snippet:

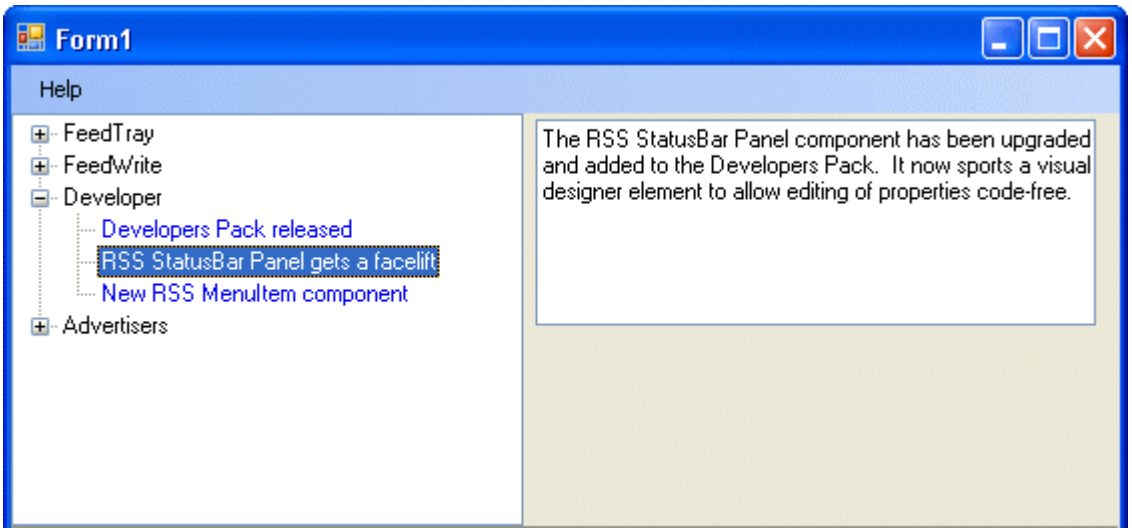
```
[C#]
using Haumohio.RSS;

       private RssDropDownButton rssDropDownButton1 = new RssDropDownButton();
...
        ((System.ComponentModel.ISupportInitialize)(this.rssMenu1)).BeginInit();

       rssDropDownButton1.Categorise = true;
       rssDropDownButton1.CompanyRegistrationEmail = "sales@haumohio.com";
       rssDropDownButton1.RssFeedUrl = "http://www.haumohio.com/haumohio.xml";

        this.statusStrip1.Items.Add( this.rssDropDownButton1 );

        ((System.ComponentModel.ISupportInitialize)(this.rssDropDownButton1)).EndIn
```

```
it();
...
       rssDropDownButton1.Fetch();

[VB]
Imports Haumohio.RSS

       Dim rssDropDownButton1 AS RssDropDownButton = New RssDropDownButton()
...
       rssDropDownButton1.BeginInit

       rssDropDownButton1.Categorise = True
       rssDropDownButton1.CompanyRegistrationEmail = "sales@haumohio.com"
       rssDropDownButton1.RssFeedUrl = "http://www.haumohio.com/haumohio.xml"

        statusStrip1.Items.Add( this.rssDropDownButton1 )

       rssDropDownButton1.EndInit
...
       rssDropDownButton1 .Fetch()
```

3.3 The RSS TreeView Component

3.3.1 Introducing The RSS TreeView

The RSS TreeView looks and acts like a normal TreeView control, but *SURPRISE!* its contents change daily. How do they do that?



The RssTreeview can be pointed at a newsfeed anywhere on the internet, or even a local file, and the articles are automatically loaded in as nodes. If the articles have been assigned a category, then the articles will be grouped under category nodes for easy access. Also, the articles that have not yet been read by the current user are highlighted by setting the text of the article's title to blue.

Once the RssTreeview has been added to a form then recipients of the different parts of the selected article can be assigned. For example, the textbox in the right hand side of the image above has been assigned as the recipient of the Description section of the selected article. This means that the contents of the textbox change as the user selects another article node in the treeview.

So, basically the RssTreeview is all you need to make your very own RSS aggregator or newsreader

in a matter of a few minutes!

Editing News Feeds

Not only does the RssTreeview read RSS files it can alter and save them without the developer knowing anything about the internal format of the RSS file.

There are a number of methods added to the RssTreeview to allow a developer to modify the content of the loaded news feed such as:

- **AddArticle**, which adds a new article to the news feed
- **RemoveNode**, which removes an article or a whole category
- **LoadValuesFromRecipients**, which amends the different parts of the selected article according to what the user has entered.
- **LoadFromFile**, which loads a local RSS file
- **SaveToFile**, which saves to a local RSS file, or even a writable internet file through http or ftp - just specify the URL of the location.

We trust this component so much that we use it ourselves as the basis for Haumohio FeedWrite, our news feed editor.

3.3.2 Configuring RSS TreeView

Configuring the RssTreeView is as simple as telling it where the news feed is, your licence details, and which controls should be used to display the relevant sections of the selected article. Once a reference to the Developer's Pack had been added to the project, adding an RssTreeView can be added as a regular control in the designer.

Configuration of the RssTreeView is performed through a few properties on the component, that may be entered using the visual designer in Microsoft Visual Studio, or just as easily with a text editor. These properties are:

URL	The URL of the news feed file itself; usually on your company's website
CompanyRegistrationEmail	Your registered email, used as a key to authorise the URL as registered. See Licensing components.
xxxRecipient	A nominated control (usually a TextBox) used for displaying and possibly editing the value of a section of the selected article. (e.g the Description, or the Author). First add the control to the form and then select it from the property's drop down list

Configuring without an IDE

The RssTreeView can be configured using the following code snippet:

```
[C#]
using Haumohio.RSS;

      private RssTreeView rssTreeView1 = new RssTreeView();
...
      this.rssTreeView1.ArticleAuthorRecipient = textBox1;
      this.rssTreeView1.ArticleDateRecipient = textBox2;
      this.rssTreeView1.ArticleDescriptionRecipient = textBox3;
      this.rssTreeView1.ArticleEnclosuresRecipient = listbox1;
      this.rssTreeView1.ArticleIDRecipient = textBox5;
      this.rssTreeView1.ArticleLinkRecipient = textBox6;
      this.rssTreeView1.ArticleTitleRecipient = textBox7;
```

```
        this.rssTreeView1.CompanyRegistrationEmail = "sales@haumohio.com;
        this.rssTreeView1.Name = "rssTreeView1";
        this.rssTreeView1.NewsfeedAuthorRecipient = textBox8;
        this.rssTreeView1.NewsfeedDescriptionRecipient = textBox9;
        this.rssTreeView1.NewsfeedImageFolderRecipient = textBox10;
        this.rssTreeView1.NewsfeedImageRecipient = pictureBox1;
        this.rssTreeView1.NewsfeedTitleRecipient = textBox11;
        this.rssTreeView1.URL = "http://www.haumohio.xml";

        this.Controls.Add( this.rssTreeView1 );

...
        rssTreeView1.Retrieve();
```

4 Installing the RSS Developer's Pack

The RssStatusPanel control uses no third party controls or libaries, and needs no special installation either on the developer's machine or on the user's machine. To install it simply add a reference to the library **Haumohio.RSS.Controls.dll** to your project.

There is no runtime licence required and the dll may be distributed with your application. Company licensing allows the components to read from a single registered news feed per instance of each component, on any number of machines around the world.

5 Licensing components

A company may use the components in this pack to read any number of *registered* news feeds. The licence is checked as follows:

1. Each component is assigned only one news feed (although multiple instances of components may read the same news feed).
2. Each component is marked as belonging to you by setting the **CompanyRegistrationEmail** property.
3. The **Fetch()** method is called on each component.
4. The component first checks with Haumohio whether the news feed is registered against the provided email address.
5. The news feed is read from the nominated location, and displayed appropriately.

If the news feed is not deemed as registered against the provided email address, the display is restricted in the following ways:

- A maximum of two (10) articles are shown per news feed
- The haumohio icon is shown rather than the news feed specified image
- A purchase link is added to the sub-menus

In all other ways the components are fully functional.

Index