Technical Notes

Questions regarding any Inmagic product should be addressed to Inmagic or your local authorized Inmagic dealer.

Inmagic
600 Unicorn Park Drive
Fourth Floor
Woburn, MA 01801 U.S.A.
Telephone: 781-938-4444 or 800-229-8398
Fax: 781-938-4446
http://www.inmagic.com

support@inmagic.com	- technical support questions
CustomerSvc@inmagic.com	- general company, product, and services questions
sales@inmagic.com	- sales, product pricing, and custom solution questions
wishlist@inmagic.com	- feature requests

9.2013

Contents

Inmagic® *Genie* Technical Notes

Inmagic® *Genie* is an integrated library system application that consists of a set of predefined textbases, forms, and business logic to support library functions such as cataloging, acquisitions, circulation of materials, serials tracking, interlibrary lending, and end user searching.

You can configure the *Genie* application to meet the specific needs of your organization. The application ships with a number of XML configuration files that contain parameters that you may modify. The names of these files suggest their area of influence: Genie.CONFIG contains general parameters, Reports.CONFIG contains content and formatting information for Summary and Full Display reports, and so forth.

While you could modify these files directly, you would be losing the ability to revert to the out-of-the-box defaults in the event that problems arise. For this reason, the *Genie* installer creates copies of these files, with a "My" prefix on each file name (for example, MyGenie.CONFIG).

We strongly recommend that you always use the My*.CONFIG files to customize the application for your purposes. As of version 2.00, the application looks for configuration changes in the My*.CONFIG files first. For example, if you want to make a change to a report, make your change in the MyReports.CONFIG file.

Do not make any changes to the configuration files without the "My" prefix. These non-My*.CONFIG files are for fall-back purposes. For example, in case you need to revert to the original settings, you can copy a non-My configuration file and rename it with the "My" prefix. If the application does not find the My*.CONFIG files, it will use the non-My*.CONFIG files.

Using the My*.CONFIG files will allow your customizations to survive future upgrades.

- **MyGenie.CONFIG** lets you change various general settings in the application. This configuration file is documented starting on page 3.
- **MyGenieFieldMaps.CONFIG** lets you change field attributes. This configuration file is documented starting on page 28.
- **MyLabels.CONFIG** lets you set up your own label formats. This configuration file is documented starting on page 39.
- **MyNavbarMenuData.CONFIG** lets you change the appearance of the *Genie* navigation bar. This configuration file is documented starting on page 45.
- **MyQueries.CONFIG** lets you change the search pages provided for the application. You can also add your own search pages if you want. This configuration file is documented starting on page 51.
- **MyReports.CONFIG** lets you change the reports provided for the application. You can also add your own reports if you want. This configuration file is documented starting on page 61.
- **MyEditScreens.CONFIG** lets you change the edit screens provided for the application. This configuration file is documented starting on page 83.
- **MyStatistics.CONFIG** lets you control what queries are performed on the Statistics page. This configuration file is documented starting on page 105.
- **MyUserData.CONFIG** lets you control what a borrower is able to see about his/her activity, such as contact information from the Borrower textbase, active loans, active reserves, etc. This configuration file is documented starting on page 110.

The configuration files reside in the ConfigFiles subfolder of your main *Genie* installation folder (for example, C:\Program Files\Inmagic\Genie\ConfigFiles).

Note: Your organization may have customized the *Genie* application, including changing the names of fields and boxes. This document refers to entities on the *Genie* graphical user interface (GUI) by their default names.

About XML

Due to the intricate coding and functions that make the *Genie* application work, the ability to change field attributes depends on your technical expertise with XML. There are rules to which you must strictly adhere. Some XML information is provided in the TextWorks online help file (for example, see the help topic "Creating Tag Names for XML Input"). Any typographical error (such as forgetting to use a closing quotation mark) will cause your changes to not take effect and could quite possibly break the application. You should always make any changes to your My*.CONFIG files. If necessary, you can refer to the non-MY*.CONFIG files to see what the original settings are.

About Resetting *Genie*

Whenever you make changes to these configuration files, you must reset *Genie* in order for your changes to appear in the application. ***It is no longer necessary to restart IIS.***

Reset *Genie* by going to the **About Genie** page while logged in as an Administrator, and clicking the **Reset** button. This removes all CONFIG file data from memory and forces the CONFIG files to be reloaded. The link to **About Genie** appears at the top of the left navigation menu on the Catalog search screen.

Contacting Inmagic and Communicating with Other Users

For help, you can contact Inmagic or your local Inmagic dealer. You can also communicate with other Inmagic users.

If you have a maintenance agreement, please have your customer ID ready, and try to be at your computer when you call. If that is not possible, note exactly what you were doing when you encountered the problem, the exact text of any error messages you received, and your software version numbers and serial numbers. (For the *Genie* serial number, click the **About Genie** link on the *Genie* navigation bar. For the TextWorks serial number, open TextWorks and choose **Help>About TextWorks**.) If you do not have a maintenance agreement, you can contact Inmagic Customer Service to purchase one.

Inmagic
600 Unicorn Park Drive, Fourth Floor
Woburn, MA 01801
U.S.A.
Tel: 781-938-4444 or 800-229-8398
Fax: 781-938-4446
http://www.inmagic.com

support@inmagic.com	- technical support questions
CustomerSvc@inmagic.com	- general company, product, and services questions
sales@inmagic.com	- sales, product pricing, and custom solution questions
wishlist@inmagic.com	- feature requests

If your message is intended for a particular person at Inmagic (for example, a Technical Support representative who is expecting the message or files), please include the name of that person in the subject and in the message.

Communicating with Other Users

You can participate in user-to-user discussions through an Inmagic forum on the Web. Note that the forums are not an official customer or technical support channel for Inmagic products. To participate in a forum, go to the Inmagic Customer Extranet at http://support.inmagic.com/downloads/extranet.html.

MyGenie.CONFIG File

Use the MyGenie.CONFIG file to change various general settings for the application.

- Adding your logo to your OPAC; see page 15.
- Adding a **Location** drop-down list to the Catalog and OPAC search pages; see page 4.
- Changing the date format for the application; see page 5.
- Configuring the currency format that displays the total cost in reports; see page 5.
- Changing the number of days after the current date for the **Needed Before** box on the Add New ILL Request page; see page 6.
- Configuring the volume and issue information for Serial records; see page 18.
- Configuring the file upload settings for Catalog records; see page 7.
- Configuring the InfoCart for your catalog and OPAC; see page 8.
- Adding links to an OPAC search screen; see page 16.
- Changing the items in the drop-down list for various fields; see page 18.
- Changing or removing the Genie product name in the heading for some screens; see page 25.
- Configuring fields copied between textbases; see page 21.
- Enabling batch update operations; see page 26.
- Adding language settings to enable culture-specific terminology or multi-lingual Genie; see page 113.
- Enabling Presto integration; see page 28.

The following sections in the MyGenie.CONFIG file are for the application and you should not need to modify them except as noted:

- **`<!-- Library-specific settings, including e-mail ->`.** When the e-mail function is set up through the Set Up E-mail Information page, the application writes to this section. Typically, you would not need to modify this section of the configuration file. The one exception would be if you wanted to disable the e-mail functionality. When you entered a mail server name on the Set Up E-mail Information page, you enabled various e-mail links in the application. If later you want to hide those e-mail links, you can only do so through the MyGenie.CONFIG file by commenting out or deleting the `<add key="MailServer" value="..." />` element and then resetting Genie. Note that you may also want to hide the e-mail items in the drop-down list for the InfoCart (see page 12) and on the Statistics report (see page 107), as they will not work once you disable the e-mail functionality.
- **`<OpacPages>`.** This section is used by the application. Typically, you would not modify this section unless instructed to by Inmagic staff or a development partner. An exception would be if you use query constraints. In this case, see the *Inmagic Genie Security Notes*.
- **`<ExpandPageList>`.** This section is used by the application. Do not modify this section unless instructed to by Inmagic staff or a development partner.

Configurable Parameters

Adding a "Location" Drop-down List to the Catalog and OPAC Search Pages

Use a query filter if you want to add a **Location** drop-down list to the bottom of the Catalog and OPAC search pages. In a multi-branch library, this feature lets the end user specify the branch or location in which to limit the search. The user's selection will be remembered in a browser cookie and used to preselect the same location for future queries.

To activate this feature, locate the following `add` element in the MyGenie.CONFIG file, change to `value="true"`, as illustrated here, and then reset *Genie*.

```
<add key="ApplyQueryFilter" value="true" ExpiresIn="180"/>
```

If you later want to remove this feature, change to `value="false"`.

Note: If the logged-in user is a member of a group with a query constraint, the **Location** drop-down list will not appear. A query constraint is hidden search criteria added to every query into the CATALOG textbase. For more about this, see the *Inmagic Genie Security Notes*.

If you want to set the number of days before the browser cookie expires, change the value in the `ExpiresIn` attribute. For example, to set the cookie to:

- Expire in 25 days: `<add key="ApplyQueryFilter" value="true" ExpiresIn="25"/>`
- Expire immediately: `<add key="ApplyQueryFilter" value="true" ExpiresIn="-1"/>`
- Never expire: `<add key="ApplyQueryFilter" value="true" ExpiresIn="0"/>`

The drop-down list is populated with the contents of the validation list associated with the CatLocation field in the Catalog textbase. You can use TextWorks to change this list (**Maintain**> **Edit Lists**).

Date Format for the Application

The *Genie* application uses the default date format of `d MMM yyyy` (day 3-letter month abbreviation 4-digit year; for example, 23 Nov 2005). This date format should work correctly for any region.

To change the date format for the application, you must change the `value` attribute in the applicable section of the MyGenie.CONFIG file and then reset *Genie*.

Note: If you want all your dates (those generated by *Genie* and Inmagic® *WebPublisher PRO*) to use a consistent format, you should also modify the `sShortDate=` parameter in the DBText.INI or InmCSrv.INI file to match, and then restart IIS. For more information about this parameter, see the TextWorks online help.

The following shows the default `add` element in the configuration file.

```
<!-- Date format used for dates generated by Genie; only .Net-compatible
   date formats can be used -->
<add key="DateFormat" value="d MMM yyyy"/>
```

If you prefer a different date format, you can change it. Select one that works in both Windows and .Net. The following example shows a date format of `M/d/yyyy`, which generates numeric dates such as 11/12/2005 (for November 12, 2005).

```
<add key="DateFormat" value="M/d/yyyy"/>
```

Note: If you specify an international numeric date with the day first (for example, d/m/yyyy), in addition to setting the `sShortDate` parameter in the DBText.INI or InmCSrv.INI file, you must also:

- Specify the appropriate culture attribute in another .CONFIG file. To learn how to set the culture in the *Genie* application, see page 112.
- Specify the `iDate=1` parameter in the [defaults] section of the DBText.INI or InmCSrv.INI file in the main *WebPublisher PRO* installation folder. For more information about this parameter, see the TextWorks online help.

Currency Format that Displays the Total Cost in Reports

You can configure the currency format that displays the total cost in reports.

To configure the currency format that displays the total cost in reports

1. Open the MyGenie.CONFIG file.
2. Find the Currency Format section and change the value for the `CurrencyFormat` attribute:

```
<!-- Currency format used for displaying total cost in reports.
   Example formats: "N", "C", "#,##0.00"  -->
<add key="CurrencyFormat" value="N" />
```

   - A value of `N` means that only numbers will appear in the total cost in reports.
   - A value of `C` means that the appropriate currency symbol for the defined culture will appear. For example, if you have culture set to `en-GB` and the currency format set to `C`, the English pound sign £ will appear in the total cost. To learn how to set the culture in the *Genie* application, see page 112.

3. Save the configuration file and reset *Genie*.

Changing or Removing the Genie Product Name

You can change or remove the Genie product name in the heading for edit, query, and report screens. To do this, locate the following element in the MyGenie.CONFIG file, and change the value or leave it empty.

```
<add key="GenieHeading"  value="Inmagic Genie"/>
```

Specifying Stop Words

Specific query screens in Genie can be configured to use alternate search syntax. Users can type words or phrases without Boolean symbols. If some of these words are stop words in the textbase (words that are not indexed), their presence in a query can cause the query to fail to find results. Use this setting in MyGenie.CONFIG to list the stop words that Genie should remove from a query before submitting it to *WebPublisher PRO*. (Note that stop words entered within a quoted phrase do not interfere with the query.)

Add this element to the **appSettings** element in MyGenie.CONFIG (substituting an appropriate list of words, separated by spaces):

```
<!-- Stop word list (used with alternate search syntax) -->
<add key="StopWords" value="a an and by for from in of the to"/>
```

Expiration of the Persistent Field Value Cookie

If you want specific field values to be automatically populated, on new record screens, with the last value saved in that field, you can add a **Persists** attribute to the field in an edit screen definition in MyEditScreens.CONFIG. The last value entered in the field is saved in a cookie on the user's local machine. By default, this cookie expires after 180 days. To change the number of days before expiration, modify this line in MyGenie.CONFIG:

```
<!--Setting for persistent fields in the edit screen. Use the ExpiresIn attribute
  to set the number of days until the cookie expires.
  To set the cookie to expire immediately, use "-1".
  To have the cookie never expire, use "0". -->
<add key="PersistsCookie" ExpiresIn="180"/>
```

Default ILL Request Due Date

The Add New ILL Request page has a **Needed Before** box, as illustrated below. The application automatically fills in this field from the setting you designate in the MyGenie.CONFIG file. Out of the box, the number is set at `30` (days). Note that you can change the date in the Needed Before box on a case-by-case basis.

To change the number of days after the current date

1. Open the MyGenie.CONFIG file.
2. Change the `value` attribute in the applicable section of the configuration file. As illustrated in the following example, the number has been changed to 60. This means, that if on March 5, 2006 you added a new ILL request, the application will insert May 4, 2006 in the **Needed Before** box of the Add New ILL Request page.

```
<!--  Number of days after the current date for the Needed by date in
    new ILL Requests-->
<add key="DaysUntilILLNeedBefore" value="60"/>
```

3. Save the configuration file and reset *Genie*.

Uploading Files

Configuring the File Upload Folders

You configure the folders to which files can be uploaded in the MyGenie.CONFIG file. You configure the links that enable file upload in the MyEditScreens.CONFIG file.

Images that are uploaded can then be viewed from links in screens and reports. Documents can be viewed in a browser if the document type is supported. PDF files can be viewed if Adobe Acrobat is installed, for example. Microsoft Word documents can be viewed if Microsoft Office is installed. The text can be extracted from documents and saved in a textbase field, for searching, if the *Importer* service is installed and set up appropriately.

Uploaded image files are placed directly into a folder for images. Uploaded documents are uploaded to one of two folders. If no text is to be extracted from the document, it is uploaded directly to a folder for documents. If text IS to be extracted, it is uploaded to a folder that the *Importer* is watching. When the text has been extracted and imported to the textbase, the document file is moved to the previously mentioned document folder.

You can also import bibliographic information for new Catalog records, in the MARC XML format provided by BookWhere.

The following shows the document and image upload section of the MyGenie.CONFIG file.

```
<!-- Document & image settings -->
<add key="ImportFolder" value="ImportFolder"/>
<add key="DocumentFolder" value="DocumentFolder"/>
<add key="ImageFolder" value="ImageFolder"/>
<add key="ToCImageFolder" value="ToCImageFolder" />
<add key="ToCImportFolder" value="ToCImportFolder" />
<add key="BookWhereFolder" value="BookWhereFolder" />
```

The `ToCImageFolder` value specifies the destination for table of contents images uploaded for serial and journal issues.

The `ImageFolder` value specifies the destination for images uploaded from other textbase edit screens.

The `ImportFolder` value specifies the initial destination for documents uploaded from the Catalog edit screen, when the Extract Text box is checked.

The `ToCImportFolder` value specifies the initial destination for table of contents documents uploaded from the Serials edit screen, when the Extract Text box is checked.

The `DocumentFolder` value specifies the folder containing documents uploaded without text extraction. This is also the folder in which the *Importer* should place documents after text has been extracted.

The `BookWhereFolder` value specifies the initial destination for BookWhere XML MARC files uploaded from a separate menu command. This is the only folder in this list where the file upload is not initiated from a textbase edit screen. It is a different type of upload, in that this file contains bibliographic data to create NEW Catalog records. The *Importer* should be set up to watch this folder and import the contents to the Catalog textbase. It doesn't matter where the uploaded file is placed after the import, as Genie won't require further access.

Image upload links can be added to any textbase edit screen except Orders, Reserves, or Items. Document upload links can be added only to Catalog or Serials edit screens.

For more information about upload links and edit screens, see Uploading Files on page 100. For information about including images or links to images or documents in reports, see the discussion of the `TreatAs` attribute on page 73.

Note: The following element in the Web.CONFIG file controls the maximum size permitted for files to be uploaded.

```
<httpRuntime maxRequestLength="102400" executionTimeout="3600"/>
```

The default value, 102400, permits file sizes up to 100 megabytes. If a user attempts to upload a file larger than the specified value, Internet Explorer will display a "Page cannot be displayed" message.

Additional Import Folders

As of version 3.4, Genie supports specification of additional folders to which you can upload files containing records to be imported into Genie textbases. In MyGenie.CONFIG, add an UploadFolders element within the configuration element. Then add an UploadFolder child element for each import profile you want to add.

Example:

```
<!-- Additional Upload Folders (for files to import) -->
<UploadFolders>
   <UploadFolder Name="UploadCSV" Folder="CsvImport" Textbase="Catalog" />
</UploadFolders>
```

The three UploadFolder attributes shown in the example are required:

- `Name` - a unique name for this UploadFolder element
- `Folder` - the folder to which the files will be uploaded (located in the main Genie folder unless a fully qualified path is supplied)
- `Textbase` - the name of the textbase into which the files will be imported

Other actions are required in order to make this feature work. For each UploadFolder, you need

- An item added to MyNavbarMenuData.CONFIG that will bring up the file upload screen. It should open the bookwhere_upload page with three added parameters.
    - Type=ImportFile
    - Name=<name of profile as specified in MyGenie.CONFIG>
    - EditScreen=<name of edit screen in MyEditScreens.config>

  Example (line breaks inserted before "&amp" for readability only):

```
<Item Id="Catalog_CSV_Upload" Text="Upload CSV File to Import"
NavigateURL="bookwhere_upload.aspx?Type=ImportFile
   &amp;Name=UploadCSV
   &amp;EditScreen=CSV_upload"
permission="Administrators Catalogers"></Item>
```

- An edit screen definition in MyEditScreens.CONFIG that defines the appearance of the file upload screen

  Example:

```
<inmg:EditScreen Name="CSV_upload" TitleForAdd="Upload CSV File to Import">
   <Toolbar Title="Data Entry Toolbar" ButtonCellWidth="260">
      <Button Name="Save" Text="Upload" />
   </Toolbar>
   <EditSection Title="Select File">
      <Field Style="font-weight: bold" . . .>%%StartTable</Field>
      <Field>%%StartRow</Field>
      <Field Display="Import File" UploadType="ImportFile"
         NewHelpText=. . .>%%FileUpload</Field>
      <Field>%%EndRow</Field>
   </EditSection>
</inmg:EditScreen>
```

- An *Importer* profile that watches the upload folder and uses appropriate settings to import the records

Some important notes:

- The Textbase attribute on the UploadFolder element is used to detect whether the logged in user has permission to edit records in the textbase (and, by extension, to import records). It does not control to which textbase the records will actually be imported – that is controlled by the textbase associated with that folder in the *Importer* settings.
- The menu item should expose the upload capability only to roles that can edit records in the specified textbase. Genie permits Administrators and Catalogers to edit any textbase, and Staff to edit any textbase except Catalog, Orders, and Items.
- The menu item should communicate to the user what will happen. "Upload File" may be less helpful than "Upload CSV file to Catalog" in the "Other" menu area, or "Upload CSV File" in the "Catalog" menu area.
- Similarly, the screen definition for the upload page should describe what will happen. This is especially important if you have more than one file upload profile, for different textbases, or for different activities (file formats, etc.) in the same textbase.

Configuring the *Importer* to Extract Document Text

The *Importer* can be set up to watch a number of import folders. Each folder is associated with a profile that specifies the textbase to update and the import settings to use. Only one target textbase and set of import parameters can be associated with a given folder. *Genie* v3.00 and later supports 3 import folders: one for the full text of Catalog documents, one for the full text of table of contents documents, and one for BookWhere files. Use the following settings for the 3 import folders.

For Catalog documents, on the Import Document Options dialog box:

- On the Document Fields tab:
  - In the Field for Text drop-down list, select **CatFullText**.
  - In the **Field for File Name** drop-down list, select **CatFileName**.
- On the Add/Replace Options tab:
  - In the If Match is Found group, select the **Replace Fields** option button.
  - In the If Match is Not Found group, select the **Reject New Record** option button.
- Import Directory: the ImportFolder value
- Move To: the DocumentFolder value

For Table of Contents documents (to be imported to the ToC textbase), on the Import Document Options dialog box:

- On the Document Fields tab:
  - In the Field for Text drop-down list, select **ToCFullText**.
  - In the **Field for File Name** drop-down list, select **ToCFileName**.
- On the Add/Replace Options tab:
  - In the If Match is Found group, select the **Replace Fields** option button.
  - In the If Match is Not Found group, select the **Reject New Record** option button.
- Import Directory: the ToCImportFolder value
- Move To: the DocumentFolder value

For BookWhere files (to be imported to the Catalog textbase), select Fielded Text Files.

- On the File Format tab:
  - Select **XML**.
  - Specify this **transform**: **BookWhere.xsl** (in the ImporterFiles folder inside the Genie installation folder).
- On the Validation tab:
  - Select **Accept Overrides** (with or without **Update Validation Lists**).
- Import Directory: the BookWhereFolder value
- Move To: doesn't matter

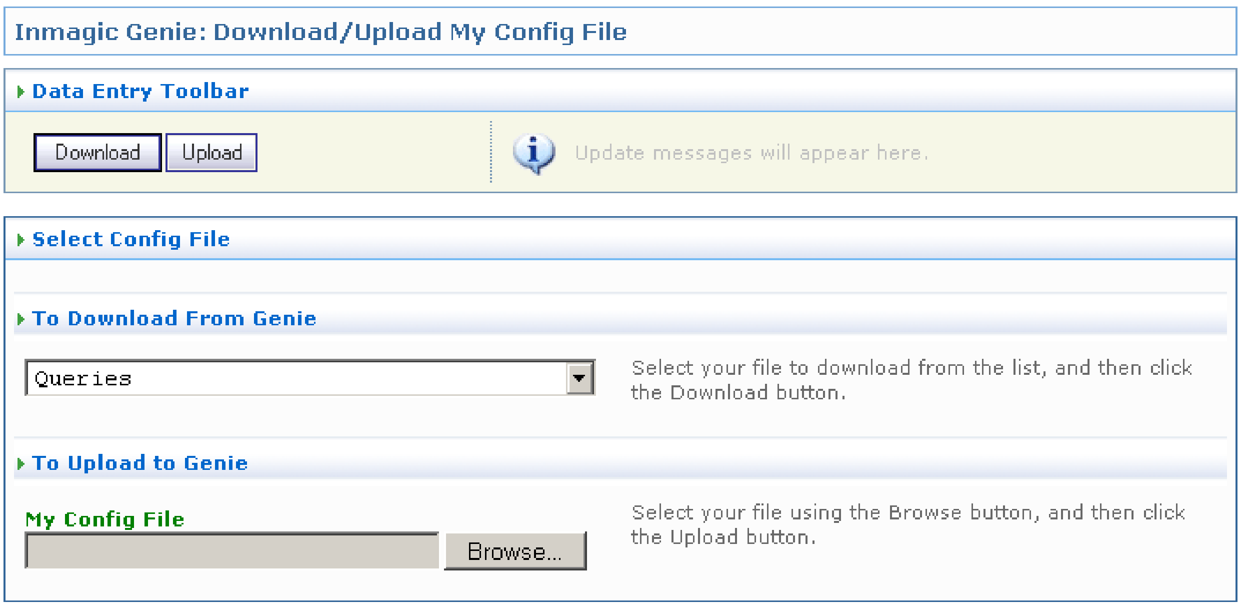
Managing My Config Files

As of version 3.5 of Genie, there is support for the ability to modify your configuration files (the files decribed in this document) by downloading a file, changing it as needed, and uploading it to a new folder in Genie named MyConfigFiles.

When opening a configuration file, Genie now looks for files in this order, and loads and uses the first one found. The example uses Genie.config for illustration, but the same pattern applies for all configuration files:

- MyConfigFiles\MyGenie.en-US.config (or alternate culture-decorated name depending on the locale you are using)
- MyConfigFiles\MyGenie,config
- ConfigFiles\MyGenie.en-US.config (or alternate culture-decorated name depending on the locale you are using)
- ConfigFiles\MyGenie,config
- ConfigFiles\Genie.config

This is the screen that supports this activity:



To enable this feature, which is not enabled by default, make these changes:

- Add an item to MyNavbarMenuData.CONFIG that will bring up the Manage Config Files screen. It should open the bookwhere_upload page with these added parameters.
    - Type=ConfigFile
    - EditScreen=<name of edit screen in MyEditScreens.config>

  Example (line breaks inserted before "&amp" for readability only):

```
<Item Id="Other_ConfigFile_Upload" Look-LeftIconUrl="DownloadUpload.gif"
      Text="Manage Config Files"
      NavigateURL="bookwhere_upload.aspx?Type=ConfigFile
      &amp;Name=UploadCfg
      &amp;EditScreen=config_upload" permission="Administrators"></Item>
```

  Note that the NavBarMenuData.CONFIG file installed with this Genie release has this menu item included, but with permission="none", which prevents it from appearing.

- Add an edit screen definition in MyEditScreens.CONFIG that defines the appearance of the management screen.

  Abbreviated example. (Full version can be copied from the v3.5 EditScreens.config and uncommented):

```
<inmg:EditScreen Name="config_upload" TitleForAdd="Download/Upload My Config File">
   <Toolbar Title="Data Entry Toolbar" ButtonCellWidth="260">
      <Button Name="Download" Text="Download" />
      <Button Name="Save" Text="Upload" />
   </Toolbar>
   <EditSection Title="Select Config File">
      <SubSection Name="DownloadFile" Title="To Download from Genie">
         <Field Style="font-weight: bold" CellSpacing="0" CellPadding="6"
              Width="100%" Align="center">%%StartTable</Field>
         <Field>%%StartRow</Field>
         <Field Display="My Config File" UploadType="ConfigFile"
              NewHelpText="Select your file to download from the list."
              UploadBoxSize="35">%%FileDownload</Field>
         <Field>%%EndRow</Field>
         <Field>%%EndTable</Field>
      </SubSection>
      <SubSection Name="UploadFile" Title="To Upload to Genie">
         <Field Style="font-weight: bold" CellSpacing="0" CellPadding="6"
              Width="100%" Align="center">%%StartTable</Field>
         <Field>%%StartRow</Field>
         <Field Display="My Config File" UploadType="ConfigFile"
              NewHelpText="Select your file and then click Upload."
              UploadBoxSize="35">%%FileUpload</Field>
         <Field>%%EndRow</Field>
         <Field>%%EndTable</Field>
      </SubSection>
   </EditSection>
</inmg:EditScreen>
```

Recommendations

- **Keep backup copies** of files you download, before and after making changes. There is no way to undo an upload. If your uploaded file has problems, from not achieveing the effect you want to breaking the XML, you can upload the previous functional version and try again. (Note that breaking the integrity of the XML syntax means that Genie will be unable to successfully read the file, and ALL of the content becomes inaccessible.)
- **Use an XML editor.** This will protect against errors that would render the file unuseable. If you mistype an element or attribute name, it won't be "seen" by Genie, but it won't break the integrity of the rest of the elements.
- **Use comments to document your changes.**

Configuring the InfoCart

Cart Actions

With the InfoCart, library staff and end users of your OPAC can select multiple Catalog records and do any of the following:

- Make a request for Catalog items via e-mail
- Send Catalog record information via e-mail to anyone, in text format or as a hypertext link
- Download the Catalog information in CSV or XML format
- Print Catalog record information
- For library staff only: Print labels for Catalog records

To configure the InfoCart, you must configure the applicable `<Action Name>` section in the `<LibraryCart>` or `<OpacCart>` section of the MyGenie.CONFIG file and then reset *Genie*. The `<LibraryCart>` section controls the InfoCart for your library staff. The `<OpacCart>` section controls the InfoCart for the end users of your library. You want to be aware of the section in which you are making your changes.

You can configure the InfoCart to:

- Configure which Catalog fields appear in InfoCart output; see page 12.
- Remove an item from the action drop-down list; see page 12.
- Change the label of an item in the action drop-down list; see page 13.
- Re-order the items in the action drop-down list; see page 13.
- For the InfoCart for the OPAC, you can specify additional attributes; see page 14.

To configure which Catalog fields appear in InfoCart output

Many `<Action Name>` sections contain a `<FieldList>` element that identifies the fields to show in the InfoCart output. Out of the box, the application provides you with some fields (where applicable). You can change the items in the `<FieldList>`.

The following shows the `<Action Name="Request" Display="Request">` element section for the InfoCart for the `<LibraryCart>` section. When a request is made for a Catalog record, the request will show the *CatTitle*, *CatSubtitle*, *CatAuthor*, and *CatCallNumber* fields as defined in the `<FieldList>` section.

```
<!-- Actions exposed to library staff in the Catalog InfoCart -->
<LibraryCart>
   <Action Name="Request" Display="Request">
      <FieldList>
         <Field>CatTitle</Field>
         <Field>CatSubtitle</Field>
         <Field>CatAuthor</Field>
         <Field>CatCallNumber</Field>
      </FieldList>
   </Action>
   ...
</LibraryCart>
```

To add another field, copy a `<Field>` element from the ones already provided and change it to another field name. For a list of the fields in the CATALOG textbase, see the MyGenieFieldMaps.CONFIG file.

To remove a field, comment out or delete the entire `<Field>` element.

To remove an item from the action drop-down list

To remove an item from the drop-down list for the InfoCart, you can remove the entire applicable `Action Name` section. For example, out of the box, there is an `Action Name="EmailText"...` section, as shown in the left column below. In the right column, the whole section has been deleted and **E-mail (Text)** will not appear in the drop-down list for the InfoCart used by library staff.

Out-of-the-box	Change, with section deleted

`<!-- Actions exposed to library staff in the Catalog InfoCart -->`
`<LibraryCart>`
`<Action Name="Request" Display="Request">`
`<FieldList>`
`<Field>CatTitle</Field>`
`<Field>CatSubtitle</Field>`
`<Field>CatAuthor</Field>`
`<Field>CatCallNumber</Field>`
`</FieldList>`
`</Action>`
`<Action Name="EmailText" Display="E-mail (Text)">`
`<FieldList>`
`<Field>CatTitle</Field>`
`<Field>CatSubtitle</Field>`
`<Field>CatAuthor</Field>`
`<Field>CatCallNumber</Field>`
`</FieldList>`
`</Action>`
`<Action Name="EmailLink" Display="E-mail (Link)">`
`<LinkText>Click here</LinkText>`
`</Action>`
`...`
`</LibraryCart>`
	`<!-- Actions exposed to library staff in the Catalog InfoCart -->`
	`<LibraryCart>`
	`<Action Name="Request" Display="Request">`
	`<FieldList>`
	`<Field>CatTitle</Field>`
	`<Field>CatSubtitle</Field>`
	`<Field>CatAuthor</Field>`
	`<Field>CatCallNumber</Field>`
	`</FieldList>`
	`</Action>`
	`<Action Name="EmailLink" Display="E-mail (Link)">`
	`<LinkText>Click here</LinkText>`
	`</Action>`
	`</LibraryCart>`

To change the label of an item (the text that appears) in the action drop-down list

To change the label of an item in the drop-down list for the InfoCart, edit the `Display` attribute for the item. In the following example, the **Request** item has been changed to **Ask the Library** for the `<OpacCart>` section:

```
<!-- Actions exposed to end users in the OPAC InfoCart -->
<OpacCart ShowCartMessage="true" CartMessage="item(s) in cart"
          ShowCartImage="true" CartImageURL="images/infocart.gif">
   <Action Name="Request" Display="Ask the Library">
      <FieldList>
         <Field>CatTitle</Field>
         <Field>CatSubtitle</Field>
         <Field>CatAuthor</Field>
         <Field>CatCallNumber</Field>
      </FieldList>
   </Action>
   ...
</OpacCart>
```

To send email without using HTML tables for formatting

A UseTable attribute can be added to the EmailText element to suppress the use of tables for formatting the records in the email. Example:

```
<Action Name="EmailText" Display="E-mail (Text)" UseTable="false">
```

To add more print reports to the InfoCart

Add reports by copying the Print action, changing the name to make it unique, specifying what should appear to identify the action, adding a **Type="Print"** attribute, and adding a **Report** attribute to specify the report to use. Example :

```
<Action Name="Print" Display="Print Items" />
<Action Name="Print1" Display="Shelf List" Type="Print" Report="CatalogShelfList" />
```

To re-order the items in the action drop-down list

To re-order the items that appear in the drop-down list for the InfoCart, move one or more items to another position in the list.

Out of the box, the **Request** item is first. In the following example for the `<OpacCart>` section, it has been moved to the third position, after the two e-mail items:

```
<!-- Actions exposed to end users in the OPAC InfoCart -->
<OpacCart ShowCartMessage="true" CartMessage="item(s) in cart"
          ShowCartImage="true" CartImageURL="images/infocart.gif">
   <Action Name="EmailText" Display="E-mail (Text)">
      <FieldList>
         <Field>CatTitle</Field>
         <Field>CatSubtitle</Field>
         <Field>CatAuthor</Field>
         <Field>CatCallNumber</Field>
      </FieldList>
   </Action>
   <Action Name="EmailLink" Display="E-mail (Link)">
      <LinkText>Click here</LinkText>
   </Action>
   <Action Name="Request" Display="Request">
      <FieldList>
         <Field>CatTitle</Field>
         <Field>CatSubtitle</Field>
         <Field>CatAuthor</Field>
         <Field>CatCallNumber</Field>
      </FieldList>
   </Action>
   ...
</OpacCart>
```

To support sending a request to a specific branch

For the Request action only, you can enable the request to be sent to any of a list of branches that will appear in a drop-down list. Add an EmailToList element and within it, an EmailTo element for each branch address. The content can be an email address by itself, or (as shown below) a name and email address in the form: `display-name <abc@def.com>`. For the latter format, encode the angle brackets as shown in the example.

```
<Action Name="Request" Display="Request">
  <FieldList>
    <Field>CatTitle</Field>
    <Field>CatCallNumber</Field>
  </FieldList>
  <EmailToList>
    <EmailTo>Woburn Library &lt;woburnlibrary@xxxx.com&gt;</EmailTo>
    <EmailTo>Burlington Library &lt;burlingtonlibrary@xxxx.com&gt;</EmailTo>
  </EmailToList>
</Action>
```

Cart Attributes

For the InfoCart for the OPAC, you can specify attributes. The following table lists the supported attributes for the `OpacCart` element.

Attribute	Required	Default	Purpose
`CartImageURL`	No	`"images/infocart.GIF"`	This is the source URL for the OPAC InfoCart icon.
`ShowCartMessage`	No	`"true"`	Displays text link for the OPAC InfoCart.
`CartMessage`	No	`"item(s) in Cart"`	This is the text displayed for the text link for the OPAC InfoCart.
`ShowCartImage`	No	`"true"`	Displays the cart image for the OPAC InfoCart.

Configuring the OPAC

Exposing Pages

By default, two pages can be accessed by OPAC users, bypassing the login screen. Anyone navigating to these pages directly will be assigned to the Public role.

These pages are listed in the `<OpacPages>` element of MyGenie.CONFIG:

```
<!-- pages that do not force the user to log in -->
<OpacPages>
  <OpacPage>opac.aspx</OpacPage>
  <OpacPage>opac_report.aspx</OpacPage>
</OpacPages>
```

Opac.aspx is the standard OPAC query screen. Opac_report.aspx is the page showing the results of a query entered on opac.aspx. Normally users would not navigate to opac_report.aspx directly, so it need not be listed in this section. It is included only to support canned queries that are directed to this page.

Patron Self-Service

Two additional pages are available to support patron self-service:

- Add **opac_checkout.aspx** to enable OPAC users to check out their own loans.
- Add **opac_loans_search.aspx** to enable OPAC users to renew their own loans.

In addition to adding these pages to MyGenie.CONFIG, you will need to provide links to these pages. An optional **Use=** parameter distinguishes between a self-service page used in a shared, **public** location (such as a terminal in the library), and one used in a **private** location (the end-user's desktop). Private pages remember the borrower in a cookie on the local machine. Public pages do not. Also, public pages clear the checkout screen after successfully processing a loan request.

Examples:

```
http://servername/inmagicgenie/opac_checkout.aspx?Use=public
http://servername/inmagicgenie/opac_loans_search.aspx?Use=private
```

If the **Use** parameter is omitted, **public** is assumed.

Adding Your Logo to Your OPAC

You can add your logo to the pages that are available for your OPAC (online public access catalog).

1. Put the logo file you want to use in the Images subfolder of your main *Genie* installation folder (for example, C:\Program Files\Inmagic\Genie\Images). The file must be in this subfolder.
2. Open the MyGenie.CONFIG file.
3. To activate this feature, remove the comment codes from the `add` element in the applicable section of the configuration file. The following shows the element in the configuration file with the comment codes removed:

```
<!-- corporate logo for OPAC screens -->
<add key="MyLogo" value="images/Inmagic_Genie_Logo-167.gif" />
```

4. Edit the file name provided out-of-the-box (`Inmagic_Genie_Logo-167.gif`) to match the file name for your logo, as shown in the following example:

```
<!-- corporate logo for OPAC screens -->
<add key="MyLogo" value="images/My_Logo.gif" />
```

5. Save the configuration file and reset *Genie*.

Optional Links on OPAC Search Screen

You can configure links to other screens available to end users on the OPAC search screen.

Use the **`OpacSearchItems`** element to add a link to an Items search screen. These attributes are available for this element:

Attribute	Req'd	Default	Purpose
`ShowSearchItemsImage`	No	`"false"`	Display an image next to the link to the items search screen.
`SearchItemsImageURL`	No	`"images/items-Search.gif"`	URL for the image to display.
`ShowSearchItemsLink`	No	`"false"`	Display a link to the items search screen.
`SearchItemsLinkText`	No	`"Search Items"`	Text of the link (if shown).

Use the **`OpacMyGenie`** element to add a link to a My Genie page, that shows information about the borrower. For this feature to work, borrowers must have login records in the GenieKey textbase, with their Borrower IDs included.

These attributes are available for this element:

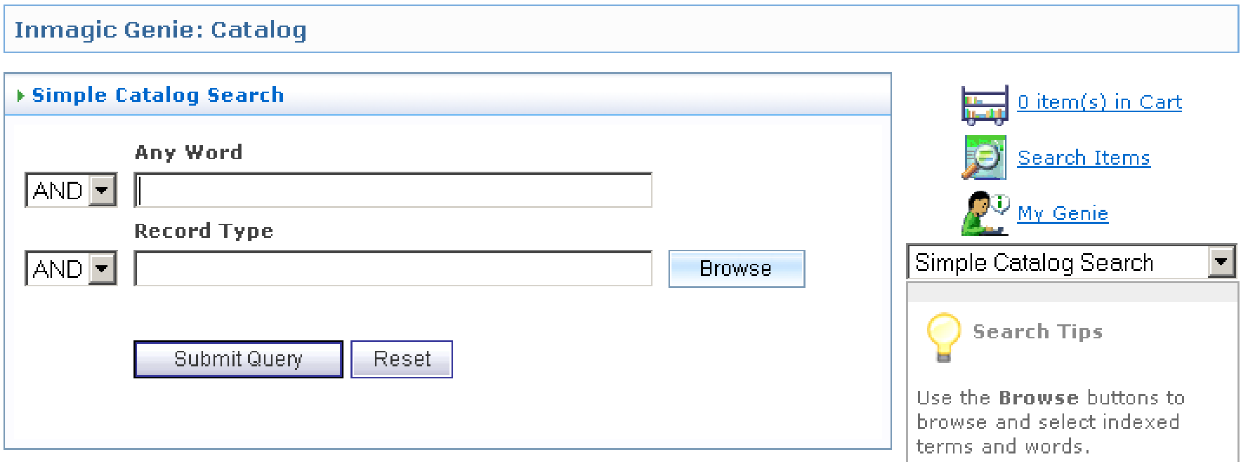
Attribute	Req'd	Default	Purpose
`ShowMyGenieImage`	No	`"false"`	Display an image next to the link to the My Genie page.
`MyGenieImageURL`	No	`"images/MyGenie.gif"`	URL for the image to display.
`ShowMyGenieLink`	No	`"false"`	Display a link to the My Genie page.
`MyGenieLinkText`	No	`"My Genie"`	Text of the link (if shown).

Use the **`OpacSearchCatalog`** element to add a link the OPAC search screen from the Items search screen. These attributes are available for this element:

Attribute	Req'd	Default	Purpose
`ShowSearchCatalogImage`	No	`"true"`	Display an image next to the link to the OPAC search page.
`SearchCatalogImageURL`	No	`"images/catalog.gif"`	URL for the image to display.
`ShowSearchCatalogLink`	No	`"true"`	Display a link to the My Genie page.
`SearchCatalogLinkText`	No	`"Search Catalog"`	Text of the link.

The following XML in the MyGenie.CONFIG file produces the links on the OPAC search screen shown below:

```
<OpacSearchItems ShowSearchItemsImage="true" ShowSearchItemsLink="true"
   SearchItemsLinkText="Search Items" SearchItemsImageURL="images/items-Search.gif"/>
<OpacMyGenie ShowMyGenieImage="true" ShowMyGenieLink="true" MyGenieLinkText="My Genie"
   MyGenieImageURL="images/mygenie.gif"/>
```



Changing the "Is in Cart" Image

The default image indicating that a Catalog record is in the InfoCart is a red check mark (✔). Some accessibility guidelines recommend that colors other than red are easier for some people to see. This release of Genie includes a black checkmark image, alt-check.gif, for this purpose. Or substitute your own image.

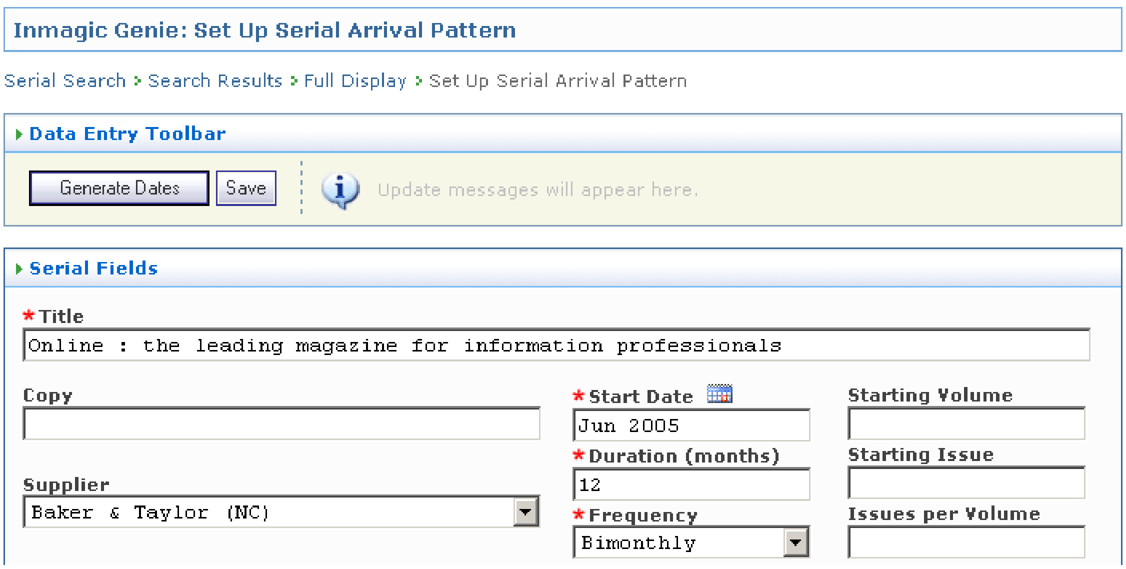
Add this element to the **appSettings** element in MyGenie.CONFIG.

```
<add key="CheckImage" value="../images/alt-check.gif" />
```

Configuring Volume and Issue Information for Serial Records

The *Genie* application has the ability to generate volume and issue information for the expected arrival dates for a Serial record. The volume and issue information is appended after a date, so that it does not interfere with searches for overdue issues. When an issue arrives and is checked in, the date, volume, and issue information can be re-arranged. You can even add the date that the issue was received.

The appearance of these items in the field entry is governed by a series of elements found in the MyGenie.CONFIG file. These elements are used with boxes on the Set Up Serial Arrival Pattern page: **Starting Volume**, **Starting Issue**, and **Issues per Volume**, as illustrated below. When you set up an arrival pattern, if you fill in these boxes on the Set Up Serial Arrival Pattern page, when you generate the issue dates, they will include volume and issue information. You will see this information (dates with volume and issue information) when you go to individually record the receipt of an issue or mark an issue as not received (by using the **Check in** and **Skip missing issue** links next to the **Issues Expected** box label).



To configure the volume and issue information for a Serial record, you must configure the following add elements in the applicable section of the MyGenie.CONFIG file and then [reset *Genie*].

```
<add key="VolumeIssueFormat" value=": v. {v}, no. {i}"/>
<add key="IssueOnlyFormat" value=": no. {i}"/>
<add key="MovedIssueFormat" value="{d} (expected), {t} (recd): {v}"/>
<add key="MissingIssueFormat" value="{d} (expected): {v}"/>
```

VolumeIssueFormat dictates the format of the volume and issue information. This is the information that will be appended to the date the issue is expected to arrive. This element contains two placeholders:

- {v} is a placeholder for the volume number.
- {i} is a placeholder for the issue number.

The text before the placeholders is the text that will show in the output. <add key="VolumeIssueFormat" value=": v. {v}, no. {i}"/> will generate entries in the **Issues Expected** box like these:

10 Sep 2005: v. 12, no. 9
10 Oct 2005: v. 12, no. 10

You should be aware of the following:

- If `VolumeIssueFormat` is not present in the MyGenie.CONFIG file, no volume/issue information is generated, and the **Starting Volume**, **Starting Issue**, and **Issues per Volume** boxes do not appear on the Set Up Serial Arrival Pattern page.
- The `<add key="VolumeIssueFormat".../>` element must begin with a unique separator character (a character that does not appear elsewhere in the inner text for the element, and also does not appear in the date that will precede it). This requirement ensures that it is clear which part of the entry is the date and which part is the volume and issue. In the example shown above, the colon (:) is the designated separator character.

`IssueOnlyFormat` dictates the format of the issue information if there is no volume in the pattern, to be appended to the date the issue is expected to arrive. This element contains one placeholder:

- `{i}` is a placeholder for the issue number.

`MovedIssueFormat` dictates the format of this entry when the issue is checked in (after you click the **Check in** link on the Edit Serial Record page), and the entry is moved to the **Issues Received** box.

- `{d}` is a placeholder for the date the issue was expected.
- `{t}` is a placeholder for the current date (when the issue is checked in).
- `{V}` [notice that this is a capital V] is a placeholder for volume and issue, exactly as it appears after the separator character in `VolumeIssueFormat` but with leading and trailing spaces removed.

If any of these placeholders is missing, the corresponding information will be omitted when the entry is moved.

`<add key="MovedIssueFormat" value="{d} (expected), {t} (recd): {V}"/>` would place these entries in the **Issues Received** box:

10 Oct 2005 (expected), 10 Oct 2005 (recd): v. 12, no. 10
10 Sep 2005 (expected), 9 Sep 2005 (recd): v. 12, no. 9

`MissingIssueFormat` dictates the format of the entry when it is moved to the **Issues Missing** box (after you click the **Skip missing issue** link). `MissingIssueFormat` uses the same three placeholders as `MovedIssueFormat`. Note that Serial claim reports include information from the **Issues Missing** box. You may need to modify these reports somewhat, depending on the format you select for this parameter.

`<add key="MissingIssueFormat" value="{d} (expected): {V}"/>` would place these entries in the **Issues Missing** box:

10 Oct 2005 (expected): v. 12, no. 10
10 Sep 2005 (expected): v. 12, no. 9

To change the format of the volume and issue information

You can change the format of the volume and issue information. For example, you can spell out "volume", "issue", and "received", as illustrated in the following example:

```
<add key="VolumeIssueFormat" value=": volume {v}, issue {i}"/>
<add key="IssueOnlyFormat" value=": issue {i}"/>
<add key="MovedIssueFormat" value="{d} (expected), {t} (received): {V}"/>
<add key="MissingIssueFormat" value="{d} (expected): {V}"/>
```

The `add` elements shown above will generate entries in the **Issues Expected** box like these:

10 Sep 2005: volume 12, issue 9
10 Oct 2005: volume 12, issue 10

In the **Issues Received** box like these:

10 Oct 2005 (expected), 10 Oct 2005 (received): volume 12, issue 10
10 Sep 2005 (expected), 9 Sep 2005 (received): volume 12, issue 9

In the **Issues Missing** box like these:

10 Oct 2005 (expected): volume 12, issue 10
10 Sep 2005 (expected): volume 12, issue 9

Things to keep in mind

- In versions of *Genie* prior to 1.2, the *Issue Due*, *Issue Received*, and *Missing Issues* fields were all strict Date fields. A strict Date field must begin with a date. If you modify `MovedIssueFormat` and/or `MissingIssueFormat` so that they do not begin with a date, ensure that the affected field definitions are Text fields or non-strict Date fields. (If you do not have textbases provided with an earlier version of *Genie*, you do not have to make these changes.)
- Case is significant in all of the elements. The `{v}` placeholder is not the same as `{V}`, and you cannot use a placeholder in an element that does not expect it. For example, `{v}` and `{i}` can only be used in `VolumeIssueFormat`.
- `<add key="MovedIssueFormat".../>` and `<add key="MissingIssueFormat".../>` are only used when you click the links over the **Issues Expected** box or when you click the **Check in** or **Skip missing issues** links on the Check In or Skip Serials report. These elements are not used if you move entries between boxes on the edit page by selecting, cutting, and pasting them. Note that if you are checking in an issue that had already been moved to the **Issues Missing** box, then you do have to use the Select, Cut, and Paste commands, then manually make any changes in the format of the information yourself.

Configuring Fields Copied between Textbases

You can specify which fields from the Orders or Catalog edit screen are copied to other textbases. Default values are shown in the illustrations below. You must specify the source field in Orders or Catalog and the target field in the target textbase, as attributes of each **Field** element.

To change the fields copied from a Serials order to the Catalog textbase when the **Create Catalog Record** button is clicked, modify this element in MyGenie.CONFIG:

```
<!-- Map the Orders fields that you want to copy to the Catalog textbase
   from the Order Serials screen. -->
<!-- If you want the CatRecordType field to be populated with a fixed text string,
   specify the value using the CatRecordType attribute for the
   SerialOrderToCatalogMap element (e.g., CatRecordType="Serial").
   Instead, you can map an Orders field to the CatRecordType field. If the
   CatRecordType field is mapped, the CatRecordType attribute will be ignored. -->
<SerialOrderToCatalogMap CatRecordType="Serial">
  <Field Order="OrdCatTitle" Catalog="CatTitle"/>
  <Field Order="OrdCatPublisher" Catalog="CatPublisher"/>
  <Field Order="OrdSerFrequency" Catalog="CatSerFrequency"/>
  <Field Order="OrdItemNumber" Catalog="CatISSN"/>
</SerialOrderToCatalogMap>
```

To change the fields copied from a non-Serials order to the Catalog textbase when the **Create Catalog Record** button is clicked, modify this element in MyGenie.CONFIG:

```
<!-- Map the Orders fields that you want to copy to the Catalog textbase from
   the Order Items screen. -->
<ItemOrderToCatalogMap>
  <Field Order="OrdCatTitle" Catalog="CatTitle"/>
  <Field Order="OrdCatPublisher" Catalog="CatPublisher"/>
  <Field Order="OrdCatAuthor" Catalog="CatAuthor"/>
  <Field Order="OrdItemNumber" Catalog="CatISBN"/>
</ItemOrderToCatalogMap>
```

To change the fields copied from an order to the Items textbase when the **Create Item Record(s)** button is clicked, modify this element in MyGenie.CONFIG:

```
<!-- Map the Orders fields that you want to copy to the Items textbase from
   the Order Items and Order Serials screens. -->
<OrderToItemMap>
  <Field Order="OrdID" Item="ItemOrdID"/>
  <Field Order="OrdCatID" Item="ItemCatID"/>
</OrderToItemMap>
```

To change the fields copied from a Catalog record to the Serials textbase when the **Create Serial Record** button is clicked, modify this element in MyGenie.CONFIG:

```
<!-- Map the Catalog fields that you want to copy to the Serials textbase from
   the Edit Catalog screen. -->
<CatalogToSerialMap>
  <Field Catalog="CatID" Serial="SerCatID"/>
  <Field Catalog="CatTitle" Serial="SerTitle"/>
</CatalogToSerialMap>
```

Changing the Items for the Drop-down List for Various Fields

For some drop-down lists in the application, you can change the items that appear in the list. You can add, edit, or delete the items in the drop-down list for the term of a loan (see instructions below); for orders, the order type (see page 23), payment method (see page 23), currency (see page 24), and order status (see page 24); and frequency for a serial (see page 25). You can also set the default value that appears in the applicable drop-down list.

To change the items in the drop-down list for the "Term of Loan" box on various pages

1. Open the MyGenie.CONFIG file.
2. Depending on what you want to do:

   - **To add a new item:** Copy a `<Term NumOfDays...>` element from the ones already provided and change it to a new loan term. You should change the value for the number of days and enter a value for how you want the item to appear in the drop-down list. The following example shows a loan term of one year, which is not provided out-of-the-box for the application. Note that wherever you place an item in the list is where it will appear on the drop-down list.

     ```
     <!-- LoanTerms -->
     <LoanTerms>
         <Term NumOfDays="0" DisplayTerm=" " />
         ...
         <Term NumOfDays="365" DisplayTerm="1 year" />
         <Term NumOfDays="-1" DisplayTerm="permanent" />
     </LoanTerms>
     ```

   - **To set a default loan term:** Add a `Default="true"` attribute to the loan term that you want to be the default choice for the drop-down list, as illustrated here.

     ```
     <LoanTerms>
         <Term NumOfDays="0" DisplayTerm=" "/>
         <Term NumOfDays="1" DisplayTerm="1 day" />
         <Term NumOfDays="3" DisplayTerm="3 days"/>
         <Term NumOfDays="7" DisplayTerm="1 week"/>
         <Term NumOfDays="14" DisplayTerm="2 weeks" Default="true"/>
         <Term NumOfDays="21" DisplayTerm="3 weeks" />
         ...
     </LoanTerms>
     ```

   - **To set the OPAC loan terms:** A separate element is available for use on the OPAC self-checkout screen, in the event that OPAC users should have different loan terms available. If no OpacLoanTerms element is found, the LoanTerms will appear on the self-checkout screen.

     ```
     <OpacLoanTerms>
         <Term NumOfDays="1" DisplayTerm="1 day" />
         <Term NumOfDays="3" DisplayTerm="3 days"/>
         <Term NumOfDays="7" DisplayTerm="1 week"/>
         ...
     </OpacLoanTerms>
     ```

     Note: The default loan term is ignored on the OPAC self-checkout screen.

   - **To delete an item:** Comment out or delete the entire `<Term NumOfDays... />` element.

3. Save the configuration file and reset *Genie*.

To change the items in the drop-down list for the "Order Type" box on the New/Edit Order page

1. Open the MyGenie.CONFIG file.
2. Depending on what you want to do:
   - **To add a new order type:** Copy an `<OrderType...>` element from the ones already provided and change it to another order type choice. The following example shows an order type of `Vendor Promotion`, which is not provided out-of-the-box for the application. Note that wherever you place an item in the list is where it will appear on the drop-down list.

     ```
     <!-- OrderTypes -->
     <OrderTypes>
         <Type>Blanket Order</Type>
         <Type>Vendor Promotion</Type>
         ...
         <Type>Standing Order</Type>
     </OrderTypes>
     ```

   - **To set a default order type:** Add a `Default="true"` attribute to the order type that you want to be the default choice for the drop-down list, as illustrated here.

     ```
     <!-- OrderTypes -->
     <OrderTypes>
         <Type>Blanket Order</Type>
         ...
         <Type Default="true">Regular</Type>
         ...
     </OrderTypes>
     ```

   - **To delete an item:** Comment out or delete the entire `<OrderType... />` element.
3. Save the configuration file and [reset *Genie*].

To change the items in the drop-down list for the "Payment Method" box on the New/Edit Order page

1. Open the MyGenie.CONFIG file.
2. Depending on what you want to do:
   - **To add a new payment method:** Copy a `<Method...>` element from the ones already provided and change it to another payment method choice. The following example shows a payment method of `Barter Exchange`, which is not provided out-of-the-box for the application. Note that wherever you place it in the list is where it will appear on the drop-down list.

     ```
     <!-- PaymentMethods -->
     <PaymentMethods>
         ...
         <Method>Barter Exchange</Method>
         <Method>Wire Transfer</Method>
     </PaymentMethods>
     ```

   - **To set a default payment method:** Add a `Default="true"` attribute to the payment method that you want to be the default choice for the drop-down list, as illustrated here.

     ```
     <!-- PaymentMethods -->
     <PaymentMethods>
         <Method Default="true">Credit Card</Method>
         ...
         <Method>Wire Transfer</Method>
     </PaymentMethods>
     ```

   - **To delete an item:** Comment out or delete the entire `<PaymentMethod... />` element.
3. Save the configuration file and [reset *Genie*].

To change the items in the drop-down list for the "Currency" box on the New/Edit Order page

1. Open the MyGenie.CONFIG file.
2. Depending on what you want to do:
   - **To add a new currency:** Copy a `<Currency...>` element from the ones already provided and change it to another currency choice. The following example shows a currency of `South African Rand - ZAR`, which is not provided out-of-the-box for the application. Note that wherever you place it in the list is where it will appear on the drop-down list.

```
<!-- Currencies -->
<Currencies>
   <Currency></Currency>
   <Currency Default="true">U.S. Dollars - USD</Currency>
   <Currency>U.K. Pounds - GBP</Currency>
   <Currency>Canada Dollars - CAD</Currency>
   <Currency>Euro - EUR</Currency>
   <Currency>Australia Dollars - AUD</Currency>
   <Currency>New Zealand Dollars - NZD</Currency>
   <Currency>South African Rand - ZAR</Currency>
</Currencies>
```

   - **To set a default currency:** Add a `Default="true"` attribute to the currency that you want to be the default choice for the drop-down list, as illustrated here.

```
<Currencies>
   <Currency></Currency>
   <Currency>U.S. Dollars - USD</Currency>
   <Currency>U.K. Pounds - GBP</Currency>
   <Currency>Canada Dollars - CAD</Currency>
   <Currency Default="true">Euro - EUR</Currency>
   <Currency>Australia Dollars - AUD</Currency>
   <Currency>New Zealand Dollars - NZD</Currency>
  </Currencies>
```

   - **To delete an item:** Comment out or delete the entire `<Currency... />` element.
3. Save the configuration file and [reset *Genie*].

To change the items in the drop-down list for the "Status" box on the New/Edit Order page

The application uses the `<Status Usage...>` attribute of the order status to determine which status to automatically select when creating a new item and receiving an item. If no `<Status Usage...>` element is specified for `new` or `receiving`, the application uses the first `<Status>` entry in the list.

We recommend that you do not delete the `Usage` attributes provided out-of-the-box.

1. Open the MyGenie.CONFIG file.
2. Depending on what you want to do:
   - **To add a new order status:** Copy a `<Status Display...>` element from the ones already provided and change it to another order status choice. The following example shows an order status of `unknown`, which is not provided out-of-the-box for the application. Note that wherever you place an item in the list is where it will appear on the drop-down list.

```
<!-- OrderStatus-->
...
<OrderStatus>
   <Status Usage="new" Display="new" />
   <Status Usage="receiving" Display="received" />
   <Status Display="closed" />
   <Status Display="backordered" />
   <Status Display="unknown" />
</OrderStatus>
```

   - **To delete an item:** Comment out or delete the entire `<Status Display... />` element.
3. Save the configuration file and [reset *Genie*].

To change the items in the drop-down list for the "Frequency" box on various pages

1. Open the MyGenie.CONFIG file.
2. Depending on what you want to do:
   - To add a new item: Copy a `<Frequency Pattern...>` element from the ones already provided and change it to a new serial frequency. You should change the value for the frequency pattern (see how to interpret the pattern codes below) and enter a value for how you want the item to appear in the drop-down list. The following example shows a frequency choice of `Biennial`, which is not provided out-of-the-box for the application. Note that wherever you place an item in the list is where it will appear on the drop-down list.

```
<!-- SerialFrequency-->
<SerialFrequency>
   <Frequency Pattern="X1" Display="" Default="true" />
   <Frequency Pattern="Y1" Display="Annual" />
   <Frequency Pattern="M6" Display="Semiannual"/>
    ...
   <Frequency Pattern="Y2" Display="Biennial"/>
   <Frequency Pattern="XX" Display="Irregular"/>
</SerialFrequency>
```

   - **To change the text that displays in the drop-down list:** Change the value for the `Display` attribute for that element.

```
<SerialFrequency>
   <Frequency Pattern="X1" Display="" Default="true" />
   <Frequency Pattern="Y1" Display="Annual" />
   ...
   <Frequency Pattern="XX" Display="No arrival pattern available"/>
</SerialFrequency>
```

   - **To set a default frequency:** Add a `Default="true"` attribute to the frequency pattern that you want to be the default choice for the drop-down list, as illustrated here.

```
<SerialFrequency>
    <Frequency Pattern="X1" Display="" />
    <Frequency Pattern="Y1" Display="Annual" />
    <Frequency Pattern="M6" Display="Semiannual"/>
    <Frequency Pattern="M3" Display="Quarterly" Default="true"/>
   ...
    <Frequency Pattern="D1 -d1" Display="Daily (no Sundays)" />
    <Frequency Pattern="D1 -d1,7" Display="Daily (no weekends)" />
</SerialFrequency>
```

   - **To delete an item:** Comment out or delete the entire `<Frequency Pattern... />` element.
3. Save the configuration file and [reset *Genie*](#).

How to interpret the pattern codes

For the `Frequency Pattern` attribute, you can use codes that begin with `Y` (for years), `M` (for months), and `D` (days). For example, `Y1` means once every year, `Y2` means every two years, `Y4` means once every four years, and so forth. The –d1,7 string on the Daily patterns shown above skips Sunday (the first day) and Saturday (the 7th day). The Set Up Serial Arrival Pattern page uses the pattern to create a series of expected arrival dates.

Enabling Batch Update Operations

Batch Modify and Batch Delete are not enabled in *Genie* as shipped, due to the potential for accidental loss of information. An example of each is present in MyGenie.CONFIG and/or Genie.CONFIG, commented out.

See page 64 for information about adding these functions to a checkbox report. Do not add Batch Modify and Batch Delete links to the same report, as records that cannot be deleted would be rendered unavailable for batch modification as well.

Enabling Batch Modify

To enable Batch Modify for a specific textbase, two things must be done. First, add a `BatchModify` element in MyGenie.CONFIG as in the following example. Second, add the function to a checkbox report (see page 64).

The following example enables batch modification in the Catalog textbase, only for the fields listed in `<Field>` elements, and only for users in the role Administrators:

```
<BatchModify Textbase="Catalog" Validation="update" SubstText="false">
   <FieldList>
      <Field>CatSubjects</Field>
      <Field>CatNotes</Field>
   </FieldList>
   <RoleList>
      <Role>Administrators</Role>
   </RoleList>
</BatchModify>
```

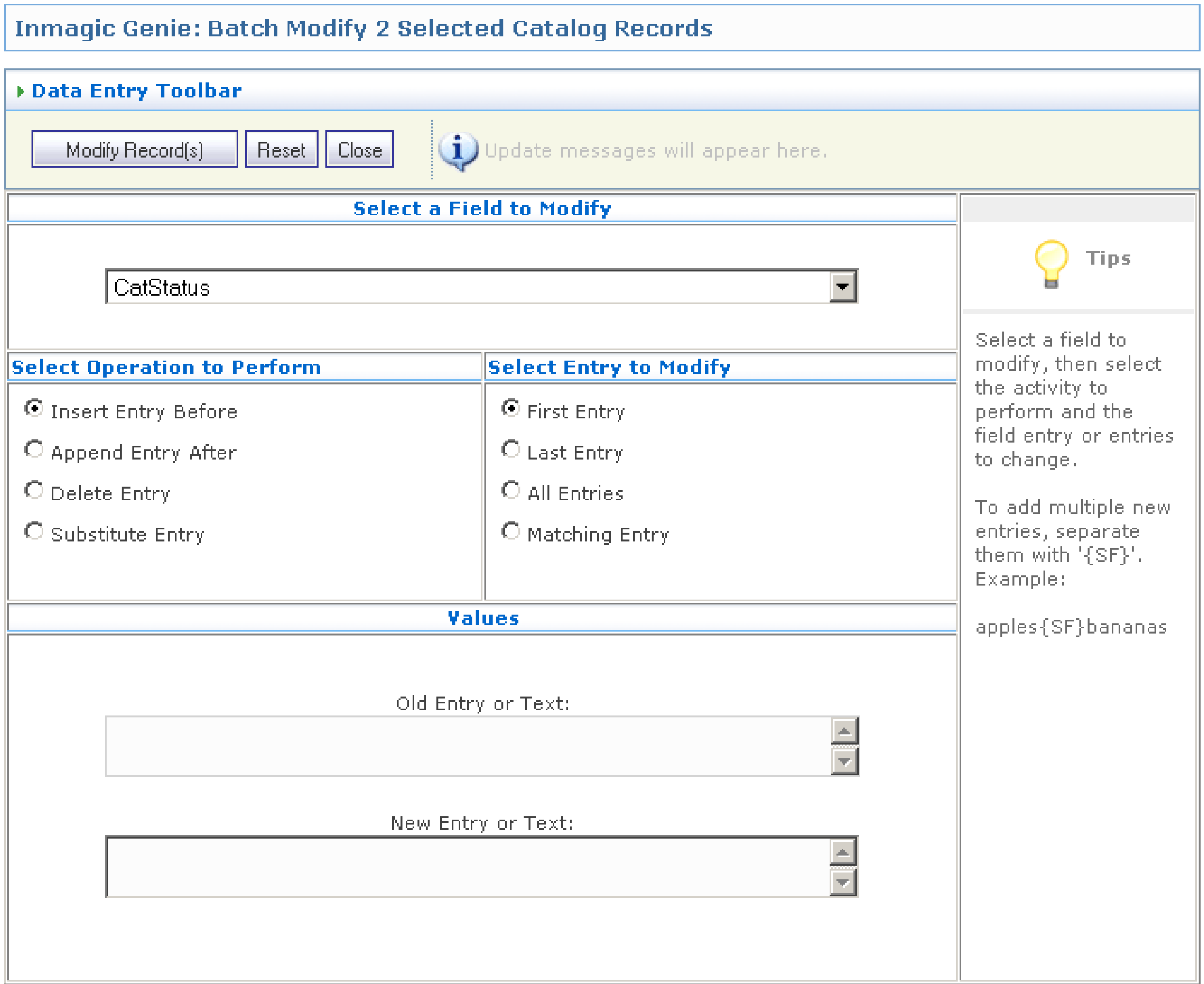
The following table lists the supported attributes for the `BatchModify` element.

Attribute	Required	Default	Purpose
`Textbase`	Yes	None	Name of the textbase for which Batch Modify will be made available to qualified users.
`Validation`	No	`"none"`	Specifies how validation will be handled (if permitted by the field definition in the textbase): `none` – reject invalid entries `override` – accept invalid entries `update` – accept invalid entries and add them to the validation list
`SubstText`	No	`"false"`	Specifies whether text substitution is available. Text substitution supports changing a word or phrase found within the contents of a field.

The `FieldList` element is required, and lists the names of the fields (in separate `Field` elements) that can be modified in this way. If a field is listed in this section that cannot be modified in this way (such as Automatic and Computed fields), that field will not appear in the list of fields that appears on the Batch Modify screen.

The `RoleList` element is also required, and lists the names of the roles (in separate `Role` elements) that can use this function in this textbase. A staff member who is not a member of a role that can access Batch Modify will not see the link to that function in a report that includes it. Note that if a specific role is not permitted to edit records in this textbase, that constraint cannot be overridden for Batch Modify.

The next page illustrates a Batch Modify screen.



Enabling Batch Delete

To enable Batch Delete for a specific textbase, , two things must be done. First, add a `BatchDelete` element in MyGenie.CONFIG as in the following example. Second, add the function to a checkbox report (see page 64).

The following example enables batch deletion in the Items textbase, only for users in the Administrators or Catalogers role:

```
<BatchDelete Textbase="Items">
   <RoleList>
      <Role>Administrators</Role>
      <Role>Catalogers</Role>
   </RoleList>
</BatchDelete>
```

The `Textbase` attribute is required.

The `RoleList` element is required, and lists the names of the roles (in separate `Role` elements) that can use this function in this textbase. A staff member who is not a member of a role that can access Batch Delete will not see the link to that function in a report that includes it. Note that if a specific role is not permitted to delete records in this textbase, that constraint cannot be overridden for Batch Delete.

Batch Delete cannot be used to remove records that are linked to by other textbases. Such records will have their checkboxes disabled in a report that includes a batch delete link. For example, a Catalog record that has associated item records or loan records cannot be deleted.

Enabling Presto Integration

Genie can be configured to copy all Catalog records to a Presto database.

Add this element to the **appSettings** element in MyGenie.CONFIG (substituting an appropriate folder name):

```
<!-- Presto synchronization -->
<add key="PrestoSyncFolder" value="C:\PrestoTarget"/>
```

This will cause Genie to write all added and updated Catalog records to the specified folder, in XML format. Records containing the IDs of deleted records are written as well. Presto would then need to be configured to watch this folder and pick up the content. XSL transforms, designed for the specific Presto database, handle the field mapping and content-type assignment.

Genie version 3.4 added the ability to copy item and loan information to Presto as well. Presto version 3.5 added support for Content Type Relationships, enabling item records to be associated with their catalog records. Active loan information can be added to the Presto item records as well.

The SyncPresto element has 3 attributes:

Attribute	Required	Purpose
`Textbase`	Yes	The name of the textbase.
`Folder`	Yes	The name of the folder into which records for Presto will be written.
`ClearOnDelete`	No	Set this to true if information in specific fields in Presto should be cleared when the textbase record is deleted. (In other words, the Presto record should NOT be deleted.) Used to remove loan information from the Presto item records instead of deleting the item record.

The SyncPresto element has one optional child element, Field. If no Field elements are specified, all fields in the Genie record will be written. Attributes for Field elements:

Attribute	Required	Purpose
`Source`	Yes	The name of the Genie field.
`Target`	No	The name of the field in the file passed to Presto. If not specified, the Genie field name is used.
`ClearOnDelete`	No	Set this to true if information in this Presto field should be cleared when the textbase record is deleted.
`ClearOnAbsence`	No	Set this to the name of a field in this textbase that, if empty, should cause the field in Presto to be cleared.
`ClearOnPresence`	No	Set this to the name of a field in this textbase that, if non-empty, should cause the field in Presto to be cleared.

Example:

```
<SyncPresto Textbase="Items" Folder="C:\PrestoTargetItems">
  <Field Source="ItemID" />
  <Field Source="ItemCatID" />
  <Field Source="ItemBarCode" />
  . . .
</SyncPresto>
<SyncPresto Textbase="Loans" Folder="C:\PrestoTargetLoans" ClearOnDelete="true">
  <Field Source="ItemID.LoanItemBC" Target="ItemID" />
  <Field Source="LoanItemBC" Target="ItemBarCode" />
  <Field Source="BorrName.LoanBorrID" Target="ItemLoanBorr"
         ClearOnPresence="LoanDateReturned" ClearOnDelete="true" />
  <Field Source="LoanDateDue" Target="ItemLoanDue"
         ClearOnPresence="LoanDateReturned" ClearOnDelete="true" />
  . . .
</SyncPresto>
```

When a loan is checked out, the synchronization process adds fields to the item record in Presto. When the item is checked back in, detected by the presence of a return date in the loan record, the loan fields in the Presto item record are cleared.

Note that if a SyncPresto element for the Catalog textbase is present, it supercedes the folder specified in the PrestoSyncFolder parameter and permits the specification of which Catalog fields to push.

Synchronization with Presto requires that Presto Services be set up to pull information from these folders.

Note that specific versions of Presto may handle synchronization with Genie and other textbases in a different way.

MyGenieFieldMaps.CONFIG File

You can make many field-specific changes to the *Genie* user interface with this file. If you change a box label, for example, you are also changing how that field is referenced in most error messages, on-page hints, browse windows, and so forth. Given this functionality, there is seldom a good reason to change the actual name of that field in the textbase.

If you do want to change the name of the field in the textbase, you will have to use TextWorks to do so, and then ensure that the corresponding parameter in the MyGenieFieldMaps.CONFIG file is changed to match.

In addition, if you want to add fields to the textbase, do so in TextWorks, then map the new fields to user-defined fields (UDFs) in the MyGenieFieldMaps.CONFIG file.

You can change either of the following attributes of a field exposed in the *Genie* application (except when noted where appropriate in this document):

- Field name (as defined in the textbase).
- Box label (as it appears if not otherwise specified on *Genie* edit and search pages)

You can change the way a textbase is identified in either of the following locations in the Genie application:

- History link
- Page header

General Instructions on How to Make Changes

When necessary, you make changes to field attributes for a textbase through TextWorks, then you make matching changes in the MyGenieFieldMaps.CONFIG file. Note that the attributes **must be** associated with the original field name.

To make changes to a *Genie* textbase through TextWorks

To make changes to a *Genie* textbase, you may have to make the changes on the desktop through TextWorks, or directly through the *Genie* application, or through both. Step 3 in the following set of steps list the changes that you can make to the *Genie* application that must be made through TextWorks and the *Genie* application. We suggest you make these changes in TextWorks first. Once you have made your changes there, you should then make the corresponding changes in the *Genie* application (see the next set of steps). For example, if you made a field required in a textbase that is also used on a page in the *Genie* application, you want to put a required-field indicator before the box label for that field, so that users can see that requirement on the page in the *Genie* application.

1. Start TextWorks. For example, from the Windows Start menu, choose **Programs>Inmagic Applications>DBTextWorks>DBTextWorks** or **Programs>Inmagic Applications>Content Server>CSTextWorks**.
2. Open the *Genie* textbase for which you want to change a field attribute.
3. Choose **Maintain>Edit Textbase Structure** to open the Edit Fields dialog box and make your field attribute changes. You can:
   - Change a field name; see page 33.
   - Change whether a field is required; see page 33.
   - Add, remove, or edit a validation list for a field; see page 33.
   - Add a field; see page 34.
4. Save your changes.
5. To reflect these changes in the *Genie* application, see the next set of steps.

To make field attribute changes reflect in the *Genie* application

1. Open the MyGenieFieldMaps.CONFIG file.
2. Within the `FieldMap` element, locate the textbase section in which you want to make field attribute changes (for example, `<inmg:FieldMap TextbaseName="Borrower" ... >`) and make your changes.
3. In the `Field` element (for example, `<Field AppName="BorrName" UserName="BorrName" BoxLabel="Name"/>` within the `FieldMap` element make your changes.
4. After you make your changes to the MyGenieFieldMaps.CONFIG file, reset *Genie*.
5. If you have not already made corresponding changes to the applicable textbase, do so now; see the previous set of steps. For example, if you added a field in the *Genie* application, you should add that field to the textbase through TextWorks.

The MyGenieFieldMaps.CONFIG file is in XML format. As installed, it contains placeholders for all of the fields that can be configured in the *Genie* textbases that are directly involved in the application, with default values supplied.

Within the guidelines as stated in this document, you can change the values in the MyGenieFieldMaps.CONFIG file to make the *Genie* application more meaningful for your end users. For each field you want to change, supply the appropriate attribute values.

The following table lists the supported attributes for the `FieldMap` element for the MyGenieFieldMaps.CONFIG file.

Attribute	Required	Default	Purpose
`Heading`	No	Textbase name	The value here is what typically appears after "Inmagic Genie:" on a page (for example, "Inmagic Genie: Interlibrary Lenders" where **Interlibrary Lenders** is the heading text).
`HistoryLink`	No	Textbase name	The value here is text (typically the meaningful name of a textbase) that is part of the label that describes a type of page (for example, the history link for a search page for the ILLENDER textbase could be "**IL Lender** Search", where **IL Lender** is the history link text).

The following table lists the supported attributes for the `Field` element for the MyGenieFieldMaps.CONFIG file.

Attribute	Required	Default	Purpose
`AppName`	Yes	None	The value here is the original field name used by the *Genie* application. **Do not change or omit the value**.
`BoxLabel`	No	The `UserName` value	The value here is the text that appears above the edit or query box for this field, if no specific box label was supplied in the edit or query screen definition. It also appears in error messages and on-page hints referencing this field. It is the default label for the field in Full Display reports, edit screens, and search pages, unless overridden by a `Display` attribute in the report, edit screen, or search page definition (separate configuration files). Do not omit this attribute unless you plan to provide `Display` parameters for all references to this field in Full Display reports, edit screens, and search pages. Also, if you omit this value, the AppName will be used to identify the field in browse windows.
`UserName`	No	Should be the same as the `AppName` attribute.	The value here is the name of the equivalent field in your textbase structure in SOAP-friendly format (for example, spaces and hyphens changed to underscores). You can change this value unless the `UserName` attribute is not present for a field in the default MyGenieFieldMaps.CONFIG file. The `UserName` attribute is only omitted for ID fields. Renaming ID fields in the textbase using TextWorks will break the *Genie* application for this textbase. ***Note:*** Because the field names almost never appear in the *Genie* graphical user interface, we do NOT recommend that you change the actual textbase field names. To change how a field name appears to the end user, change the `BoxLabel` attribute instead, as this text is used to identify the field in error messages related to the field and Browse windows for the field, as well as box labels if no alternate label is supplied.

Example of the BORROWER section

```
<inmg:FieldMap TextbaseName="Borrower" Heading="Borrowers"
       HistoryLink="Borrower">
    <Field AppName="BorrID" />
    <Field AppName="BorrName" UserName="BorrName" BoxLabel="Name" />
    <Field AppName="BorrNumber" UserName="BorrNumber"
       BoxLabel="Borrower Number" />
    ...
</inmg:FieldMap>
```

Example of the BORROWER section with changes

In the `BorrName` element, the `Username` and `BoxLabel` attributes have been changed.

```
<inmg:FieldMap TextbaseName="Borrower" Heading="Borrowers" HistoryLink="Borrower">
    <Field AppName="BorrID" />
    <Field AppName="BorrName" UserName="Full_Name" BoxLabel="Full Name (last, first)"
    />
    ...
</inmg:FieldMap>
```

Changing a Field Name

You can change a field name if you really must. To do so, you have to change the field name in the textbase with TextWorks (see page 30) and specify that change in the `UserName` attribute for the applicable textbase section in the MyGenieFieldMaps.CONFIG file to reflect the change in the application (see page 31).

Example of changing a field name

```
<Field AppName="LoanItemBC" UserName="LoanItemBarCode"
   BoxLabel="Item Bar Code" />
```

Note that a field name submitted in XML must be valid, both in terms of the *WebPublisher PRO* product and in terms of the *Genie* application. The field name must be a valid TextWorks field name. When you change the field name in the textbase using TextWorks (see page 30), it will ensure that the field name is compliant. When you then specify the new name in the MyGenieFieldMaps.CONFIG file, you should modify the name as instructed in the TextWorks help topic "Creating Tag Names for XML Input" (for example, by replacing spaces with underscores). Note that *Genie* uses the `SOAPFormat ON` setting, so be sure to use the appropriate instructions. Also be sure to specify the appropriate case. If a textbase field name is "CatEditor," do not set the `UserName` attribute in the MyGenieFieldMaps.CONFIG file to "cateditor." XML is case-sensitive.

Important! You cannot change the following field names because they are used as unique identifiers in the *Genie* textbases. They also omit the `UserName` attribute to indicate that they cannot be renamed.

- *CatID* in CATALOG
- *BorrID* in BORROWER
- *ILLID* in ILL
- *ILnID* in ILLENDER
- *ItemID* in ITEMS
- *LoanID* in LOANS
- *OrdID* in ORDERS
- *ResID* in RESERVES
- *SerID* in SERIALS
- *SuppID* in SUPPLIER
- *ToCID* in TOC

Including a Required-field Indicator

In previous versions of *Genie*, attributes in MyGenieFieldMaps.CONFIG controlled whether a required field indicator (red asterisk) appeared by the box label in an edit screen.

In *Genie* version 3.00 and later, required field indicators are specified in MyEditScreens.CONFIG. See the `Required` attribute on page 93.

If you make a field Required using TextWorks, we recommend that you indicate this to the user on the edit screen. If you do not, and the user leaves the field empty, Web*Publisher PRO* will reject the record when it is saved.

Including a Browse Link

In previous versions of *Genie*, attributes in MyGenieFieldMaps.CONFIG controlled whether a browse link appeared by the box label in an edit screen.

In *Genie* version 3.00 and later, browse links are specified in MyEditScreens.CONFIG. See the `BrowseField` and `ShowBrowse` attributes in the table on page 92.

If you remove a validation list from a field using TextWorks, remove the browse link for that field on the edit page. If you do not, and the user clicks that link, an empty browse window will appear.

If you add a validation list to a field using TextWorks, we recommend that you add a browse link for that field in the edit screen, to assist the user doing data entry.

Adding a New Field

You can add a new field to the *Genie* application. Adding a new field involves TextWorks and assorted Genie .CONFIG files.

- Use TextWorks to add a new field to a *Genie* textbase (see page 30).
- Use the MyGenieFieldMaps.CONFIG file to add a <Field> element for the new field in the applicable textbase section of the configuration file, and map it to the new field in the textbase (see below).
- Edit MyEditScreens.config to add the new field to the edit screen for the textbase.
- Edit MyReports.CONFIG to add the new field to report(s) of your choice.

Example of adding a new field

If you wanted to enter birthdates for your borrowers, you would use TextWorks to add a *BorrBirthdate* field to the BORROWER textbase. (The "Borr" prefix is recommended, but not required.) Then you would add a corresponding new Field element to the BORROWER section of the MyGenieFieldMaps.CONFIG file and map it to the applicable field in the textbase through the UserName attribute, as illustrated in this example:

```
<Field AppName="BorrUdf01" UserName="BorrBirthdate" BoxLabel="Date of Birth"/>
```

Note: As of *Genie* 2.00, you can control where user-defined fields appear on the Summary and Full Display reports by adding them to your MyReports.CONFIG file. The same is true for the MyQueries.CONFIG file.

As of *Genie* 3.00, you can control where and how they appear on edit screens by adding them to your MyEditScreens.CONFIG file.

Tip! The *Genie* application shows only those fields that are the most commonly used. For example, the record creation and modification dates in each textbase are not listed in MyGenieFieldMaps.CONFIG. To have such fields appear in reports, add them to MyGenieFieldMaps.CONFIG and MyReports.CONFIG, as described above.

About Deleting Fields

Do not delete any of the textbase fields exposed in the *Genie* application. Fields not involved in the business logic can be repurposed or removed from reports and screens. Deleting a textbase field may interfere with the functioning of the application.

Changing a Box Label

You can change a box label for the *Genie* application if you want. You can specify the label for a specific box in an edit screen, for example, or you can omit the label and default to the BoxLabel attribute specified in MyGenueFieldMaps.CONFIG. The BoxLabel attribute is also used to identify the field when browsing indexes and validation lists.

Many characters may be used for box labels. However, note that several punctuation characters should not be used. For example, the following will cause error messages:

- " Will cause an error message to appear at the very top of the first *Genie* page you access and will cause all the changes in the MyGenieFieldMaps.CONFIG file to be ignored. If you must use this symbol, use its encoded equivalent, "
- & Will cause an error message to appear at the very top of the first *Genie* page you access and will cause all the changes in the MyGenieFieldMaps.CONFIG file to be ignored. If you must use this symbol, use its encoded equivalent, &
- < Will cause .Net error messages. Even if you use its encoded equivalent, the result will not be as you want.

Changing the Text in a Heading and/or History Link for a Page

You make these changes in your MyGenieFieldMaps.CONFIG file (see page 31).

You can change the text in a heading and/or history link for a page. The heading text is what typically appears after "Inmagic Genie:" on a page. The history link is directly under the heading area and shows how you got to where you are. The history link text is part of the label that describes a type of page (for example, the history link for a search page for the catalog would be "**Catalog** Search", where **Catalog** is the history link text.)

You may want to change the text for a heading and/or history link to use more understandable terms for your end users. In the absence of a setting in the MyGenieFieldMaps.CONFIG file, the application uses the name of the textbase. For example, in the case of the CATALOG textbase, using the name of the textbase may suffice. However, in the case of the ILLENDER textbase, we have provided you with more easily understandable text for the heading and history link, which you can change as you want.

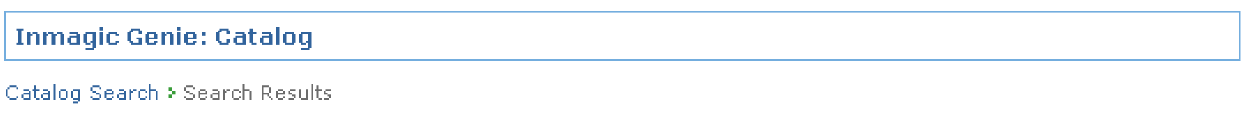
Change or add a `Heading` attribute and/or `HistoryLink` attribute to the `inmg:FieldMap` element for a textbase section, as illustrated for the ILLENDER textbase:

```
<inmg:FieldMap TextbaseName="ILLender" Heading="Interlibrary Lenders"
   HistoryLink="IL Lender">
```

Examples of headings and history links

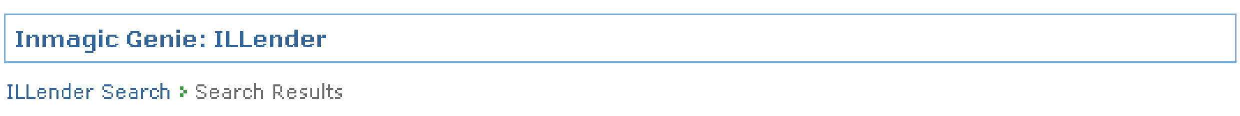
The following illustration shows the heading and history link for the pages for the CATALOG textbase in the MyGenieFieldMaps.CONFIG file. No changes have been made to the section of the configuration file because the name of the textbase can be used in both the heading name (Inmagic Genie: **Catalog**) and the history link text (**Catalog** Search > Search Results) with clarity.

```
<inmg:FieldMap TextbaseName="Catalog">
```



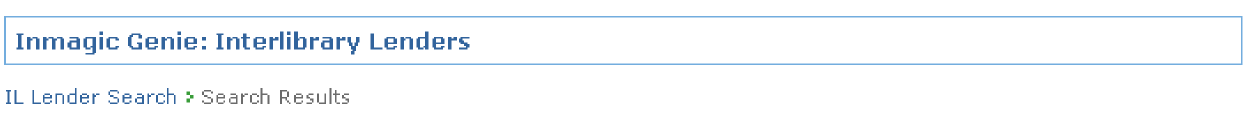
The following illustration shows the heading and history link for the ILLENDER textbase without any `Heading` and/or `HistoryLink` attributes. Users may not recognize what function they are dealing with because the heading (Inmagic Genie: **ILLender**) and the history link text (**ILLender** Search > Search Results) may not be easily understandable.

```
<inmg:FieldMap TextbaseName="ILLender">
```



The following illustration shows the heading and history link for the ILLENDER textbase in the MyGenieFieldMaps.CONFIG file with `Heading` and/or `HistoryLink` attributes. End users should more easily recognize what function they are dealing with because the heading (Inmagic Genie: **Interlibrary Lenders**) and the history link text (**IL Lender** Search > Search Results) are more easily understandable.

```
<inmg:FieldMap TextbaseName="ILLender" Heading="Interlibrary Lenders"
   HistoryLink="IL Lender">
```



More Information About Field Attributes in TextWorks

The following notes apply to field attributes that have not been specifically mentioned.

- You can change field type and special filing rules for most fields. The exceptions are noted here.
- You must not change the field type of an Automatic Number field, or the field type or associated field for a Link field.
- You can add a word or term index to a field, but removing an index may have an impact on flexibility when searching.
- You should communicate changes in the validation rules for a field to anyone who will be editing records over the Web. You may be able to use the `BoxLabel` attribute in the MyGenieFieldMaps.CONFIG file for this. For example:

  ```
  <Field AppName="BorrEmail" BoxLabel="E-mail (must be unique)"...
  ```

MyLabels.CONFIG File

Use the MyLabels.CONFIG file to set up label formats to suit your organization's needs. Out of the box, the MyLabels.CONFIG file is set up to print labels for Catalog records (accessed from the InfoCart) and to print routing slips for Serial records on label or card stock (accessed from the report drop-down list). For Catalog records, the MyLabels.CONFIG file is set up with label formats to print University Products labels and Gaylord labels, and spine labels. Note that besides including primary textbase fields, you can also include secondary textbase fields in your label forms.

Note that you can design a label for any *Genie* textbase. For an example, see page 63.

You can make changes to the following levels in the MyLabels.CONFIG file:

- **Label.** At the `Label` element level you create your label settings; see page 40.
- **TextArea.** At the `TextArea` element level you specify the areas where label information can be printed; see page 42.
- **Field.** At the `Field` element level you specify the field data to be printed in the defined text area; see page 43.

To create a new label format, we suggest you copy a label format already defined in the MyLabels.CONFIG file and change the copied elements group as necessary for your new label format. For your new label format:

1. For the `Label` element (see page 40), change the `Name` attribute, which is required and must be unique for every label format, and the other attributes, as necessary. Note that, typically, this name does not appear in the user interface. The name should avoid characters that need to be escaped in XML, including ampersand, angle brackets, and extended characters. It may be useful to the limit use of punctuation in label names. Case is also significant.
2. Change the attributes for the `TextArea` element (see page 42) as needed to specify the areas where label information can be printed.
3. Change the attributes in the `Field` element (see page 43) as needed to specify the field data to be printed in the defined text area. You can include primary and secondary textbase fields. If you use secondary textbase fields, be sure to specify a `Form` attribute that references a textbase form that includes all of the primary and secondary textbase fields needed to print the label.

Making Label-level Changes for Label Formats

The following table lists the supported attributes for the `Label` element. See the examples after the table.

Attribute	Required	Default	Explanation
`AddField`	No	None	Used in the Serials Routing List label to append a Borrower field value after the corresponding entry in the SerRouteList field.
`AddFieldSeparator`	No	None	Used with AddField to specify the text to appear between the SerRouteList entry and the Borrower field. For example, an ellipsis.
`BottomMargin`	No	`"0.5"`	Sets the bottom print margin. ***Tip!*** You can use this attribute to adjust the page break.
`Form`	No	None	Use to identify the name of a form inside the textbase that *WebPublisher PRO* uses as a source of fields to be returned in the XML result. If the `Form` attribute is absent, or if the named form does not exist in the textbase, all primary textbase fields are returned, and the records are sorted according to the textbase default sort specification. If the `Form` attribute is present and the named form exists, all primary and secondary textbase fields included in that form are returned in the XML, and the records are sorted by the named form's compulsory sort specification (if any). **Use the `Form` attribute if your label includes any secondary textbase fields, and/or should be sorted in an order other than the default order**. To improve performance, you can also use this attribute if your report includes a small subset of the fields in the textbase, and you want to avoid having fields you do not need returned in the XML result.
`ItemCallNumberRequired`	No	`"false"`	If set to `"true"`, only the labels with nonempty *ItemCallNumber* field values will be printed.
`LabelSetHeight`	Yes	None	Sets the height of a label set.
`LabelSetWidth`	Yes	None	Sets the width of a label set.
`LeftMargin`	No	`"0.5"`	Sets the left print margin.
`MarkAsRouted`	No	`"false"`	Used for the SERIALS textbase. If set to `"true"`, a **Mark As Routed** button will be displayed on the label print page (for example, Print Routing Notices). After printing, you would click the **Mark As Routed** button to batch clear the route dates. See also the `MarkAsRouted` attribute for the MyReports.CONFIG file on page 66.
`Name`	Yes	None	Unique name for the label type. It should be the same as the one listed in the LibraryCart/Action section of the MyGenie.CONFIG file (see page 8) or the `ReportList` section in the MyReports.CONFIG file.
			Continued...

Attribute	Required	Default	Explanation
`NumberAcross`	Yes	None	Sets the total number of label sets across horizontally.
`ShowBorder`	No	`"false"`	Displays border of label set and text area. It should always be `false` for normal use. ***Tip!*** You can set this attribute to `"true"` to help with troubleshooting any problems.
`TopMargin`	No	`"0.5"`	Sets the top print margin.
`Unit`	No	`"inch"`	The applicable values are `"inch"` and `"cm"`.

Example of the `Label` element for Gaylord labels

```
<inmg:Label Name="Gaylord" LabelSetWidth="7.5" LabelSetHeight="1.25"
   Unit="inch" NumberAcross="1" LeftMargin="0.5" TopMargin="0.5"
   BottomMargin="0.5" ShowBorder="false">
<TextArea Name="Box1" Left="0.2" Top="0.1" Width="0.6" Height="1.10">
<Field Name="CatCallNumber" TreatAs="WordList"/>
...
</inmg:Label>
```

Example of including secondary textbase fields in a label form and printing labels

The following is an example of how to enable the printing of Supplier address labels from the Serial module where fields from the secondary textbase SUPPLIER are used.

1. In the MyLabels.CONFIG file, add the following coding. Note that `Form="PrintSerialsClaim"` identifies a form you would create in the textbase that includes all the necessary fields.

```
<inmg:Label Name="SerialLabels" LabelSetWidth="3" LabelSetHeight="1.5" Unit="inch"
   NumberAcross="2" LeftMargin="0.25" TopMargin="0.25" BottomMargin="0.25"
   ShowBorder="false" Form="PrintSerialsClaim">
<TextArea Name="Box1" Left="0.1" Top="0.1" Width="3" Height="1.5">
<Field Name="SuppAddress.SerSuppID" Footer="&lt;br&gt;" Style="font-size:12pt" />
<Field Name="SuppCity.SerSuppID" Footer=", " Style="font-size:12pt" />
<Field Name="SuppState.SerSuppID" Footer=" " Style="font-size:12pt"/>
<Field Name="SuppPostalCode.SerSuppID" Style="font-size:12pt" />
</TextArea>
</inmg:Label>
```

2. In the MyReports.CONFIG file, include the label form `SerialLabels` in the `ReportList` area of the MyReports.CONFIG file, as illustrated here:

```
<!-- ** Note: Use special TextbaseName="SerialsClaim" to specify the reports
listed
   in serial claim related reports -->
<inmg:ReportList TextbaseName="SerialsClaim">
<Report Name="SerialsClaim" Display="Serials Claim"/>
<Report Name="PrintSerialsClaim" Display="Print Serials Claim"
   PageName="batch_print.aspx"/>
<Report Name="SerialLabels" Display="Address Labels" Type="Label" />
</inmg:ReportList>
```

3. Save the configuration files and reset *Genie*.

Once set up, to print Supplier address labels for the Serial claim letters, a user would do the following:

1. Choose **Serials**>**Claim Missing Issues** and click the **Submit Query** button.
2. Select **Print Serials Claim** from the drop-down list to produce the reports.
3. Select **Address Labels** to produce the addresses and print your labels.

Making TextArea-level Changes for Label Formats

The following table lists the supported attributes for the `TextArea` element. See the examples after the table.

Attribute	Required	Default	Explanation
`Align`	No	`"Left"`	Sets the horizontal alignment. Supported values are: `"Left"`, `"Center"`, and `"Right"`.
`Height`	Yes	None	Sets the height of the text area. ***Tip!*** If a line is cut off in the middle when you print the label set, shorten the height to eliminate the half-line.
`ItemInfo`	No	None	Identifies which fields from the ITEMS textbase you want to include on a book label. Note that this can only be used with labels associated with the CATALOG textbase. ***Tip!*** If a label report includes an `<ItemInfo>` section and you use a query constraint or a query filter to print branch-specific labels, the *ItemLocation* field must be populated with the correct location in order for the label for that item to print. A query constraint is hidden search criteria added to every query into the CATALOG textbase. For more about query constraints, see the *Inmagic Genie Security Notes*. For more about query filters, see "Configurable Parameters Adding a "Location" Drop-down List to the Catalog and OPAC Search Pages" on page 4.
`Left`	Yes	None	Sets the absolute position at left within the label set.
`Name`	Yes	None	Unique name for the text area.
`Top`	Yes	None	Sets the absolute position at top within the label set.
`Width`	Yes	None	Sets the width of the text area.

Example of the `TextArea` element for Gaylord labels

```
<inmg:Label Name="Gaylord" LabelSetWidth="7.5" LabelSetHeight="1.25"
   Unit="inch" NumberAcross="1" LeftMargin="0.5" TopMargin="0.5"
   BottomMargin="0.5" ShowBorder="false">
   ...
   <TextArea Name="Box2" Left="1.49" Top="0.1" Width="0.6" Height="1.10">
      <Field Name="CatCallNumber" TreatAs="WordList"/>
   </TextArea>
   ...
</inmg:Label>
```

Example of the `TextArea` element with an `ItemInfo` section.

Include an `ItemInfo` section within a `TextArea` element when you want to include fields from the ITEMS textbase in a book label.

```
<inmg:Label Name="Gaylord" LabelSetWidth="7.5" LabelSetHeight="1.25"
   Unit="inch" NumberAcross="1" LeftMargin="0.5" TopMargin="0.5"
   BottomMargin="0.5" ShowBorder="false">
   <TextArea Name="Box1" Left="0.2" Top="0.1" Width="0.6" Height="1.10">
   <Field Name="CatCallNumber" TreatAs="WordList"/>
   <ItemInfo>
      <Field Name="ItemVolume" Header="&lt;br&gt;"/>
      <Field Name="ItemIssue" Header="&lt;br&gt;"/>
   </ItemInfo>
   </TextArea>
   <TextArea Name="Box2" Left="1.49" Top="0.1" Width="0.6" Height="1.10">
      <Field Name="CatCallNumber" TreatAs="WordList"/>
   </TextArea>
   ...
</inmg:Label>
```

Making Field-level Changes for Label Formats

The following table lists the supported attributes for the `Field` element. See the example after the table.

Attribute	Required	Default	Explanation
`CssClass`	No	None	CSS class, defined in Print.CSS, applied to field contents, headers, and footers. The Print.CSS file resides in the main *Genie* installation folder.
`Footer`	No	None	Used to add extra characters after the field contents.
`Header`	No	None	Used to add extra characters and before the field contents.
`Name`	Yes	None	Should be the same as the `AppName` attribute (see page 32) defined in the MyGenieFieldMaps.CONFIG file for the field names in the CATALOG textbase.
`Separator`	No	`"; "`	Used when a field has multiple entries.
`Style`	No	None	HTML style applied to field contents, headers, footers, and separators.
`TreatAs`	No	None	Has value of "WordList". `TreatAs="WordList"` is used to display each word of the CATALOG Call Number field on a separate line. Use a non-breaking space character (Alt+0160) to keep two words (for example, "Vol. 1") together. Note that if the CATALOG Call Number field has more than one entry, only the first call number will be used and all other entries will be ignored.
`UpperCase`	No	As returned from the textbase	If you use a value of `"true"`, then all text will print out in uppercase.

Example of the `Field` element for Gaylord labels

```
<inmg:Label Name="Gaylord" LabelSetWidth="7.5" LabelSetHeight="1.25"
   Unit="inch" NumberAcross="1" LeftMargin="0.5" TopMargin="0.5"
   BottomMargin="0.5" ShowBorder="false">
<TextArea Name="Box1" Left="0.2" Top="0.1" Width="0.6" Height="1.10">
<Field Name="CatCallNumber" TreatAs="WordList"/>
...
</TextArea>
...
<TextArea Name="Box5" Left="5.35" Top="0.1" Width="2.11" Height="1.10">
<Field Name="CatAuthor" Footer="&lt;br/&gt;"/>
<Field Name="CatTitle" UpperCase="true"/>
</TextArea>
</inmg:Label>
```

Special Tags

The following table lists predefined special tags that you can use in the label configuration. You enter predefined tags as a field element's inner text in the MyLabels.CONFIG file. See the example below.

Tag	Explanation
`%%Label`	Displays any text and/or HTML tags that are not related to a textbase field. See the example below.

Supported Functions for Label Printing

The following table lists the functions available for label printing that you can use in the label configuration. See the example below.

Tag	Explanation
`CountEntries`	Returns the number of entries.
`Date`	Displays today's date. See the example below.
`FirstEntry`	Returns the first entry.
`From`	Displays "From" information. Uses information supplied by the *Genie* Set Up E-mail Information page.
`NumberOfEntries(x)`	Used with fields that have multiple entries. Specifies *x* number of entries to be displayed. By default, all entries will be displayed.

Example with special tag and supported function for label configurations

```
<inmg:Label Name="LabelRoute3on8.5x11" LabelSetWidth="7.5"
   LabelSetHeight="3.45" Unit="inch" NumberAcross="1"
   LeftMargin="0.5" TopMargin="0.3" BottomMargin="0.35" ShowBorder="false"
   MarkAsRouted="true">
   <TextArea Name="Box1" Left="0.0" Top="0.5" Width="7.3" Height="0.40"
   Align="Center">
      <Field Display="Routing List for " Style="font-size:12pt">%%Label</Field>
      <Field Name="SerTitle" Footer="&amp;nbsp;" Style="font-weight:bold;font-
      size:12pt"/>
      <Field Name="SerCopy" Header="(" Footer=")" Style="font-size:12pt"/>
   </TextArea>
   <TextArea Name="Box2" Left="0.0" Top="1.0" Width="3.5" Height="0.22">
      <Field Name="SerDateToRoute" Header="Date of Issue: " Style="font-size:12pt"
      UseFunction="FirstEntry"/>
   </TextArea>
   <TextArea Name="Box3" Left="4.0" Top="1.0" Width="3.2" Height="0.22">
      <Field Display="Today's date: " Style="font-size:12pt">%%Label</Field>
      <Field UseFunction="Date" Style="font-size:12pt"/>
   </TextArea>
   <TextArea Name="Box4" Left="0.0" Top="1.31" Width="3.5" Height="2.25">
      <Field Name="SerRouteList" Separator="&lt;br/&gt;" Style="font-size:12pt"/>
   </TextArea>
   <TextArea Name="Box5" Left="4.0" Top="1.31" Width="3.2" Height="2.25">
      <Field Display="Please pass this issue on within 3 days. Cross out your name
      after
      reading, or pass on unread." Style="font-size:12pt">%%Label</Field>
   </TextArea>
</inmg:Label>
```

MyNavbarMenuData.CONFIG File

Use the MyNavbarMenuData.CONFIG file to change the look of the *Genie* navigation bar to suit your organization's needs.

The following topics are covered in this section:

- "Changing the Text and/or Image for a Navigation Bar Button and its Command Links" next.
- "Hiding a Navigation Bar Button and All its Command Links" on page 47.
- "Hiding a Command Link for a Navigation Bar Button" on page 48.
- "Re-ordering the Navigation Buttons and/or Command Links" on page 49.
- "Making a Command Link Not Appear Depending on Login" on page 50.

Changing the Text and/or Image for a Navigation Bar Button and its Command Links

1. Find the group of XML elements in the MyNavbarMenuData.CONFIG file that describes the section of the navigation bar that you want to change. Out of the box, the elements for the navigation bar buttons and their command links are listed in the following order: Catalog, Borrower, Loan, Serial, Supplier, Order, ILL, and Other.
2. The first `Item` element for a menu group (for example, `Item Id="Catalog"...`) corresponds to the navigation bar button.
   - To change the text for the button, change the text after the `Text` attribute in the applicable element group.
   - To change the image for the button, change the name of the GIF file for the `Look-LeftIconUrl` attribute. Note that the GIF file must reside in the Images subfolder of the main *Genie* installation folder (for example, C:\Program Files\Inmagic\Genie\Images). To make an image not appear, you can delete the image file name.
3. The Item elements nested within the menu element describe the command links for the particular menu.
   - To change the image associated with a command link, change the name of the GIF file for the `Look-LeftIconUrl` attribute in that applicable nested element.
   - To change the text for a command link, change the text for the `Text` attribute.
4. Save the configuration file.

Note that the navigation bar has a default width of 200 pixels. If you lengthen any of the menu text enough, 200 pixels may not be enough and the menu will look odd. You can increase the width using a `Width` attribute on the `<Items>` element, for example:

```
<Items Width="210">
```

Example of the Catalog menu group

The following shows the elements group for the Catalog menu and its command links and images, before you make any changes to them.

```
<Item Id="Catalog" Text="Cataloging" itemtype="Menu"
       DefaultSubItemLookId="Level2ItemLook" SubGroupCssClass="Level2Group"
       SubGroupItemSpacing="5" Expanded="false" Look-LeftIconUrl="Catalog.gif">
    <Item Id="Catalog_CartLink" Look-LeftIconUrl="infocart.gif"
       Text="item(s) in cart" NavigateURL="catalog_cart.aspx "
       permission="Administrators Catalogers Staff"></Item>
    <Item Id="Catalog_Search" Look-LeftIconUrl="Book-Search.gif" Text="Search Catalog"
       NavigateURL="catalog_search.aspx"
       permission="Administrators Catalogers Staff"></Item>
    <Item Id="Catalog_Add" Look-LeftIconUrl="Card-Edit.gif"
       Text="Add New Catalog Item" NavigateURL="catalog_edit.aspx"
       permission="Administrators Catalogers"></Item>
</Item>
```

Example of the Catalog group, customized

The following shows the elements group for the Catalog menu and its command links with customizations. In the top-level `Item` element, both the text for the navigation bar button and the image for the button have been changed. In the first nested `Item` element within the elements group, only the image has been changed. In the second and third nested `Item` elements, only the text for the command links has been changed. In addition, in the third nested `Item` element, the image file name has been deleted so that only the command link text appears on the navigation bar.

```
<Item Id="Catalog" Text="Catalogue" itemtype="Menu"
       DefaultSubItemLookId="Level2ItemLook" SubGroupCssClass="Level2Group"
       SubGroupItemSpacing="5" Expanded="false" Look-LeftIconUrl="Bookshelves.gif">
    <Item Id="Catalog_CartLink" Look-LeftIconUrl="Blue_cart.gif"
       Text="item(s) in cart" NavigateURL="catalog_cart.aspx "
       permission="Administrators Catalogers Staff"></Item>
    <Item Id="Catalog_Search" Look-LeftIconUrl="Book-Search.gif"
       Text="Search Catalogue" NavigateURL="catalog_search.aspx"
       permission="Administrators Catalogers Staff"></Item>
    <Item Id="Catalog_Add" Look-LeftIconUrl="" Text="Add New Catalogue Item"
       NavigateURL="catalog_edit.aspx"permission="Administrators Catalogers"></Item>
    <Item Id="Item_Search" Look-LeftIconUrl="items-Search.gif" Text="Search Items"
       NavigateURL="items_search.aspx"
       permission="Administrators Catalogers Staff"></Item>
</Item>
```

Hiding a Navigation Bar Button and All its Command Links

1. Find the elements in the MyNavbarMenuData.CONFIG file that contains the section of the navigation bar that you want to hide. Out of the box, the elements for the navigation bar buttons and their command links are listed in the following order: Catalog, Borrower, Loan, Serial, Supplier, Order, ILL, and Other.
2. Insert comment codes at the very beginning of this section and at the very end of the section.
3. Save the configuration file.

Example of hidden Interlibrary Loans navigation bar button and its command links

The following shows the elements group for the Interlibrary Loans navigation bar button and its command links commented out so that this menu does not appear on the navigation bar. The begin comment code <!-- is at the beginning of the section being commented out. The end comment code --> is at the end of the commented-out section.

```
<!-- <Item Id="ILL" Text="Interlibrary Loans" itemtype="Menu"
             DefaultSubItemLookId="Level2ItemLook"
             SubGroupCssClass="Level2Group" SubGroupItemSpacing="5"
             Expanded="false" Look-LeftIconUrl="interlibrary-loans.gif">
          <Item Id="ILL_Search" Look-LeftIconUrl="search-loans.gif"
             Text="Search Loans" NavigateURL="ill_search.aspx"
             permission="Administrators Catalogers Staff"></Item>
          <Item Id="ILL_Add" Look-LeftIconUrl="add-loan.gif" Text="Add New Loan"
             NavigateURL="ill_edit.aspx"
             permission="Administrators Catalogers Staff"></Item>
          <Item Id="ILLender_Search" Look-LeftIconUrl="search-lenders.gif"
             Text="Search Lenders" NavigateURL="illender_search.aspx"
             permission="Administrators Catalogers Staff"> </Item>
          <Item Id="ILLender_Add" Look-LeftIconUrl="add-lender.gif"
             Text="Add New Lender" NavigateURL="illender_edit.aspx"
             permission="Administrators Catalogers Staff"></Item>
</Item> -->
```

Hiding a Command Link for a Navigation Bar Button

1. Find the elements group in the MyNavbarMenuData.CONFIG file that contains the command link you want to hide. Out of the box, the elements group for each navigation bar button and its command links are listed in the following order: Catalog, Borrower, Loan, Serial, Supplier, Order, ILL, and Other.
2. Insert comment codes at the very beginning of the `Item` element for the command link and at the very end of the element.
3. Save the configuration file.

Example of hidden Order Items command link from Orders navigation bar button

The following shows the elements group for the Orders navigation button and its command links. One link is the Order Items command link, which has been commented out so that it does not appear in the navigation bar. The begin comment code <!-- is at the beginning of the specific `Item` element being commented out. The end comment code --> is at the end of the commented out element.

```
<Item Id="Order" Text="Orders" itemtype="Menu"
              DefaultSubItemLookId="Level2ItemLook"
              SubGroupCssClass="Level2Group" SubGroupItemSpacing="5"
              Expanded="false" Look-LeftIconUrl="orders.gif">
       <Item Id="Orders_Search" Look-LeftIconUrl="search-orders.gif"
              Text="Search Orders" NavigateURL="orders_search.aspx"
              permission="Administrators Catalogers Staff"></Item>
       <Item Id="Order_Serial" Look-LeftIconUrl="order-serials.gif"
              Text="Order Serials" NavigateURL="orders_edit.aspx?Serial=true"
              permission="Administrators Catalogers"></Item>
<!--   <Item Id="Order_Items" Look-LeftIconUrl="order-items.gif"
              Text="Order Items" NavigateURL="orders_edit.aspx"
              permission="Administrators Catalogers"></Item> -->
       <Item Id="Receiving_Order" Look-LeftIconUrl="receiving-orders.gif"
              Text="Receiving Orders" NavigateURL="orders_Search.aspx?Receiving=true"
              permission="Administrators Catalogers"></Item>
              ...
</Item>
```

Re-ordering the Navigation Buttons and/or Command Links

You can re-order the navigation bar buttons and/or command links by moving the applicable elements group (for a button and its command links) or a particular nested element for a command link. Wherever you move the command link is where it will appear on the navigation bar, whether you move it within the same button, move it under another button, or move it to the top level of the navigation bar. For example, if you move a command link between two elements groups (say the Catalog and Borrower groups), the command link is inserted as a button link between the two other buttons you inserted in between. You can move the command link to the top of the navigation bar and the bottom of it. As another example, you may want to move the Log Out function from under the Other button to the top level of the navigation bar (see example below).

1. To re-order a navigation bar button and its command links, select the elements group in the MyNavbarMenuData.CONFIG file that contains the configuration elements for the button and its command links. Out of the box, the element groups for the navigation bar buttons and their command links are listed in the following order: Catalog, Borrower, Loan, Serial, Supplier, Order, ILL, and Other. Use the Cut command to cut the section, go to the place where you want to insert the navigation bar button and its command links, and use the Paste command to insert the section where you want it.
2. To re-order a link, select the `Item Id` nested element for the link and use the Cut and Paste commands.
3. Save the configuration file.

Example of a link moved to top level of the navigation bar

Out of the box, the link to log out resides under the Other button. The following shows the elements group for the Other button group with the nested `Item` element for the logout function outside of the Other button group. Because of its placement in the example, a Log Out button is added to the navigation bar after the Other button. This gives you top-level access to the logout function.

```
<Item Id="Other" Text="Other" itemtype="Menu"
       DefaultSubItemLookId="Level2ItemLook" SubGroupCssClass="Level2Group"
       SubGroupItemSpacing="5" Expanded="false" Look-LeftIconUrl="other.gif">
    <Item Id="Other_Help" Look-LeftIconUrl="about.gif" Text="Help Topics"
       NavigateURL='javascript:OpenHelpWindow(0,"")'
       permission="Administrators Catalogers Staff"></Item>
    <Item Id="Other_EMail" Look-LeftIconUrl="letter.gif"
       Text="Set Up E-mail Information" NavigateURL="ConfigurePage.aspx"
       permission="Administrators Catalogers"></Item>
    <Item Id="Other_Logins" Look-LeftIconUrl="padlock.gif"
       Text="Manage Security &amp; Logins" NavigateURL="key_search.aspx"
       permission="Administrators"></Item>
    <Item Id="Other_Statistics" Look-LeftIconUrl="views.gif" Text="Statistics"
      NavigateURL="statistics_report.aspx"
       permission="Administrators Catalogers Staff"></Item>
    <Item Id="Other_BookWhere_Marc_XML" Look-LeftIconUrl="marc.gif"
      Text="Upload BookWhere XML" NavigateURL="bookwhere_upload.aspx"
       permission="Administrators Catalogers"></Item>
</Item>
<Item Id="Other_LogOut" Look-LeftIconUrl="Logout.gif" Text="Log Out"
       NavigateURL="logout.aspx"
       permission="Administrators Catalogers Staff"></Item>
```

Making a Command Link Not Appear Depending on Login

You can use the MyNavbarMenuData.CONFIG file to control if a command link appears on the navigation bar depending on the login. The `permission` attribute for each command link string in the MyNavbarMenuData.CONFIG file is for convenience and is not a security function. For example, if you do not want your library staff to be able to add new lenders for interlibrary loans, you can make the Add New Lender command link under the Interlibrary Loans button on the navigation bar not appear for those who log on as library staff while keeping it for those who log on as administrators or catalogers.

1. Find the elements group in the MyNavbarMenuData.CONFIG file that contains the command link that you want to make not appear depending on the login. Out of the box, the elements group for the navigation bar buttons and their command links are listed in the following order: Catalog, Borrower, Loan, Serial, Supplier, Order, ILL, and Other.
2. In the `permission` attribute for the command link element, delete the value that corresponds to the login type.
3. Save the configuration file.

Example of making a command link not appear depending on login

The following shows the elements group for the Interlibrary Loans button group. Every command link is set to show all command links when you log in as an administrator (`admin`). However, when you log in as library staff (`staff`), you will not see the Add New Lender command link.

```
<Item Id="ILL" Text="Interlibrary Loans" itemtype="Menu"
      DefaultSubItemLookId="Level2ItemLook"
      SubGroupCssClass="Level2Group"
      SubGroupItemSpacing="5" Expanded="false"
      Look-LeftIconUrl="interlibrary-loans.gif">
   <Item Id="ILL_Search" Look-LeftIconUrl="search-loans.gif"
      Text="Search Loans" NavigateURL="ill_search.aspx"
      permission="Administrators Catalogers Staff"></Item>
   <Item Id="ILL_Add" Look-LeftIconUrl="add-loan.gif" Text="Add New Loan"
      NavigateURL="ill_edit.aspx"
      permission="Administrators Catalogers Staff"></Item>
   <Item Id="ILLender_Search" Look-LeftIconUrl="search-lenders.gif"
      Text="Search Lenders" NavigateURL="illender_search.aspx"
      permission="Administrators Catalogers Staff"></Item>
   <Item Id="ILLender_Add" Look-LeftIconUrl="add-lender.gif"
      Text="Add New Lender" NavigateURL="illender_edit.aspx"
      permission="Administrators Catalogers"></Item>
   ...
</Item>
```

MyQueries.CONFIG File

The MyQueries.CONFIG file defines all the search pages (for example, the Catalog Search page) provided for the application. You can use this configuration file to set up more than one search page per textbase and specify which should be the default search page for a textbase, add or remove a query box from a search page, and configure the Search Tips box that appears on each search page.

The following topics are covered in this section:

- "Setting Up More Than One Search Page Per Textbase" below.
- "Adding or Removing Query Boxes From a Search Page" on page 53.
- "Configuring the Search Tips Box" on page 59.

Setting Up More Than One Search Page Per Textbase

You can use the MyQueries.CONFIG file to set up one or more search pages per textbase. When you do so, you can also decide which one should be the default search page that appears when a user accesses the associated textbase for searching. Once you have set up this functionality for searching a textbase, a drop-down list will appear on the search page from which the user can then select any other search page associated with the textbase.

Note that out-of-the-box, for the CATALOG textbase, there are two search pages available for library staff and another two for your OPAC users. For library staff:

- `<inmg:QueryScreen Name="catalog_search"  Title="Catalog Search"...>`
- `<inmg:QueryScreen Name="simple_catalog_search"  Title="Simple Catalog Search"...>`

For your OPAC users:

- `<inmg:QueryScreen Name="opac_search"  Title="Catalog Search"...>`
- `<inmg:QueryScreen Name="simple_opac_search"  Title="Simple Catalog Search"...>`

To set up one or more search pages per textbase

1. Open the MyQueries.CONFIG file.
2. Go to the `QueryScreen` element section for the search page to which you want add another search page.
3. We suggest you copy a `QueryScreen` element section already defined for the textbase and change it as necessary (such as delete and add `QueryField` elements, and add to and delete from them).

4. Change the `Name` and `Title` attributes for your new `QueryScreen` element section. The following example shows the search pages defined for the CATALOG textbase, with `QueryScreen Name="simple_catalog_search"`, which is a second search page provided in the MyQueries.CONFIG file, highlighted in boldtype below.

```
<inmg:QueryScreen Name="catalog_search"   Title="Catalog Search"...>
   <QueryField Type="QueryField"  Display="Any Word"...>
   <QueryField Type="QueryField"  Display="Title"  CssClass="SearchTextInput"...>
   <QueryField Type="QueryField"  Display="Author"  CssClass="SearchTextInput"...>
   <QueryField Type="QueryField"  Display="Subject"
   CssClass="SearchTextInput"...>
   <QueryField Type="QueryField"
   CssClass="SearchTextInput"
      LabelCssClass="SearchLabel"  ShowBrowse="true"  ShowBoolean="true"
      FieldList="CatRecordType"/>
   <QueryField Type="QueryField"...>
   <QueryField Type="QueryField"...>
   <QueryField Type="ButtonRow"  SubmitButtonText=" Submit Query "...>
      <SearchTips ShowTips="true" ShowTitle="true" ShowIcon="true"...>
</inmg:QueryScreen>
```

```
<inmg:QueryScreen Name="simple_catalog_search"  Title="Simple Catalog Search"...>
   <QueryField Type="QueryField"  Display="Any Word"...>
   <QueryField Type="QueryField"                       CssClass="SearchTextInput"
      LabelCssClass="SearchLabel" ShowBrowse="true"  ShowBoolean="true"
      FieldList="CatRecordType"/>
   <QueryField Type="ButtonRow"  SubmitButtonText="  Submit Query "
      ResetButtonText=" Reset " CssClass="submit_button"/>
</inmg:QueryScreen>
```

5. Add the search page you created in the `<inmg:QueryScreens TextbaseName="Catalog">` section at the end of the MyQueries.CONFIG file. Use `<inmg:QueryScreens TextbaseName="Opac">` if the search page is for the OPAC catalog. Use `<inmg:QueryScreens TextbaseName="OpacItems">` if the search page is for OPAC items. If you want your new search page to be the default search page that appears when the end user opens the applicable textbase for searching, add it to the beginning of the list for that textbase section, as shown in the following example:

```
<!-- query screen options of a textbase -->
   <inmg:QueryScreens TextbaseName="Catalog">
      <inmg:QueryScreen Name="simple_catalog_search"></inmg:QueryScreen>
      <inmg:QueryScreen Name="catalog_search"></inmg:QueryScreen>
   </inmg:QueryScreens>
   <inmg:QueryScreens TextbaseName="Supplier">
      <inmg:QueryScreen Name="simple_supplier_search"></inmg:QueryScreen>
      <inmg:QueryScreen Name="supplier_search"></inmg:QueryScreen>
   </inmg:QueryScreens>
```

Tip! If you want a search page you created to be the only search page available for a textbase, do not include any other `QueryScreen` element section in the `QueryScreens` element section for that textbase.

6. Save the configuration file and reset *Genie*.

Query Retention

As of version 3.5, you can return to the query screen with its search criteria retained by clicking the "breadcrumb" link near the top of page. Criteria are not retained if you click a menu item to get to the query screen. If you prefer not to retain the search criteria when clicking the link for a specific query screen, add a new attribute on the query screen definition. See the description of ClearQuery below.

Adding or Removing Query Boxes From a Search Page

You can use the MyQueries.CONFIG file to add or remove one or more query boxes from a search page.

To add or remove a query box on a search page

1. Open the MyQueries.CONFIG file.
2. Find the `<inmg:QueryScreen...>` section that defines the search page that you want to change. For example, `<inmg:QueryScreen Name="borrower_search"...>` for the Borrower Search page, `<inmg:QueryScreen Name="catalog_search"...>` for the Catalog Search page, and so forth. To learn about this element, see "The `QueryScreen` Element" on page 54. The following example shows this element for the Borrower Search page.

```
<inmg:QueryScreen Name="borrower_search" Title="Borrower Search"
        BrowseButtonText="Browse" BooleanAnd="AND" BooleanOr="OR"
        BooleanNot="NOT" >
    <QueryField Type="QueryField" Display="Any Word"
        CssClass="SearchTextInput" LabelCssClass="SearchLabel"
        ShowBrowse="false" ShowBoolean="true" FieldList="BorrNumber
        | BorrName | BorrDepartment | BorrTelephone | BorrFax | BorrEmail
        | BorrAddress | BorrMailStop | BorrCity | BorrState | BorrPostalCode
        | BorrCountry"/>
    ...
</inmg:QueryScreen>
```

3. Depending on what you want to do:
   - To add a new query box to a search page: Copy a `<QueryField Type="QueryField"...>` element from the ones already provided and change the applicable attributes. For example, change the `Display` and `FieldList` attributes. To learn about this element, see "The `QueryField` Element" on page 55. The following example shows adding a query box for telephone number that is not provided out-of-the-box for the Borrower Search page.

```
<inmg:QueryScreen Name="borrower_search" Title="Borrower Search"
        BrowseButtonText="Browse" BooleanAnd="AND" BooleanOr="OR"
        BooleanNot="NOT" >
    <QueryField Type="QueryField" Display="Any Word"
        CssClass="SearchTextInput" LabelCssClass="SearchLabel"
        ShowBrowse="false" ShowBoolean="true" FieldList="BorrNumber
        | BorrName | BorrDepartment | BorrTelephone | BorrFax | BorrEmail
        | BorrAddress | BorrMailStop | BorrCity | BorrState
        | BorrPostalCode | BorrCountry"/>
    <QueryField Type="QueryField" CssClass="SearchTextInput"
        LabelCssClass="SearchLabel" ShowBrowse="true"
        ShowBoolean="true" FieldList="BorrName"/>
    <QueryField Type="QueryField" CssClass="SearchTextInput"
        LabelCssClass="SearchLabel" ShowBrowse="true"
        ShowBoolean="true" FieldList="BorrNumber"/>
    <QueryField Type="QueryField" Display="Telephone Number"
        CssClass="SearchTextInput" LabelCssClass="SearchLabel"
        ShowBrowse="true" ShowBoolean="true"
        FieldList="BorrTelephone"/>
    ...
</inmg:QueryScreen>
```

   Note that wherever you place the `QueryField` element is where the query box will appear on the search page.

   - To remove a query box: Delete the entire `<QueryField Type="QueryField".../>` element.
4. Save the configuration file and [reset *Genie*](#).

The QueryScreen Element

In the MyQueries.CONFIG file, the `QueryScreen` element identifies the search page and contains its configuration settings in the format string (list of attributes). The following table lists the supported attributes for this element.

Attribute	Required	Default	Explanation
`AlternateSyntax`	No	“false”	Specifies whether alternate search syntax is used with queries on this screen. See below.
`BooleanAnd`	No	"AND"	Identifies Boolean AND in the AND, OR, NOT drop-down list that appears to the left of the query boxes.
`BooleanOr`	No	"OR"	Identifies Boolean OR in the AND, OR, NOT drop-down list that appears to the left of the query boxes.
`BooleanNot`	No	"NOT"	Identifies Boolean NOT in the AND, OR, NOT drop-down list that appears to the left of the query boxes.
`BrowseButtonText`	No	"Browse"	Use to set the text that appears on the [out-of-the-box] **Browse** button that appears to the right of the query box.
`ClearQuery`	No	"false"	Set to true if you don't want the search criteria retained when you return to this query screen from a report or cart page by clicking the breadcrumb link.
`Heading`	No	Textbase name	The value for this attribute is applied to the page heading. The page heading is the top heading on a (search) page (for example, "Inmagic Genie: Borrowers").
`Name`	Yes	None	If you do not specify an `<inmg:QueryScreens...>` section to specify the list of query screens for a particular textbase: **Do not change this attribute value.** Use to identify the search page name. The value for this attribute matches the ASPX page name. If you do specify an `<inmg:QueryScreens...>` section: This value is the name of the search page.
`Title`	No	Textbase name	The value for this attribute is applied to the (search) page title and the title bar of the browser window. A page title appears below a page heading, for example, "Borrower Search". The value for this attribute is also used in the query screen drop-down list. However, the default in this case is the `Name= value` in the query screen definition, rather than the textbase name.

Alternate Search Syntax

A query screen that uses this syntax allows the user to enter search terms using a syntax similar to the one used by Google:

- Boolean symbols are not required, and if present, they are ignored
- there is an assumed AND between words typed in a query box
- two words separated by OR (must be upper case) means find either word (or both)
- a series of words within quotes is treated as a phrase search
- exclude a word by immediately preceding it with a hyphen

The ability to perform range searches and proximity searches using standard syntax has been retained.

If you add this feature to a query screen, provide appropriate screen tips rather than accepting the default screen tips, which describe the standard (original) syntax. See also discussion of stop words.

The `QueryField` Element

The `QueryField` element lets you specify the query boxes that you want to appear in the search page. The element contains its configuration settings in the format string. You can specify one or more fields from either the primary or a linked textbase to search. The textbase fields listed in the `FieldList` attribute should be the same as the `AppName` attribute (see page 32) specified in the MyGenieFieldMaps.CONFIG file for any field that is to be searched.

The following table lists the supported attributes for the `QueryField` element.

Attribute	Required	Default	Explanation
`BooleanValue`	No	`"AND"`	Initial state of the Boolean drop-down list or hidden value.
`CssClass`	No	For a query box, use `"SearchTextInput"`. For a check box, use `"SearchCheckBox"`.	CSS class, defined in Genie.CSS, used for the text users type or paste in a query box or the text associated with a check box. The Genie.CSS file resides in the main *Genie* installation folder.
`DefaultValue`	No	None	Use to supply text that will appear in this box initially. See also IsHidden.
`Display`	No	The `BoxLabel` attribute in the MyGenieFieldMaps. CONFIG file (see page 32) for the first field in the `FieldList` attribute.	If this value is included, it has a higher priority than the `BoxLabel` attribute (see page 32) in the MyGenieFieldMaps.CONFIG file. Typically you would want to set this attribute if your query box searches more than one field (for example, out of the box, the **Any Word** query box on the Borrower Search page [as shown in the example on page 58] and the Catalog Search page).
`FieldList`	Yes	None	Use to determine which fields are to be searched by that query box. If more than one field is to be searched, separate each field name with a pipe character (|) as shown in the example on page 58 for the Borrower Search page. The field names in the `FieldList` entry must match the `AppName` attribute (see page 32) for the textbase as listed in the MyGenieFieldMaps.CONFIG file. Note that if a field to be searched is a field in a secondary textbase, then you must specify it in the format SecondaryFieldName .PrimaryLinkFieldName. See the example on page 58 with `"BorrName.LoanBorrID"`, where `BorrName` refers to the *BorrName* field in the secondary textbase (BORROWER), and `LoanBorrID` to the Link field in the primary textbase (LOANS) that links to the secondary textbase. Note also that if there is no value for this attribute (in other words, `FieldList=""`), this query box will not appear on the search page.
`IsHidden`	No	`"false"`	Set to "true" to hide the query box. Use with DefaultValue to include a hidden condition with the query.
`LabelCssClass`	No	`"SearchLabel"`	CSS class, defined in Genie.CSS, used for the query box label. The Genie.CSS file resides in the main *Genie* installation folder. *Continued…*

Attribute	Required	Default	Explanation
`ShowBoolean`	No	`"true"`	Set to `"true"` to display a drop-down list with Boolean AND, OR, NOT before a query box or "false" to hide the drop-down list. Note that in the `QueryScreen Name="loans_search"` section, because of the intricacies of the Loan Search page, the Boolean drop-down list will always appear for the **Loan Date Returned** query box. Therefore, the `ShowBoolean` attribute does not appear in the MyQueries.CONFIG file for this query box. Even if you were to add `ShowBoolean="false"` for this query box, it would be ignored.
`ShowBrowse`	No	`"true"`	Set to `"true"` to display a **Browse** button after a query box or "false" to hide the **Browse** button.
`ShowCalendar`	No	`"false"`	Set to `"true"` to display a calendar icon next to the box label for a box on a query page. Once clicked, a calendar control appears from which a user can select a date.
`Type`	No	`"QueryField"` Other values are: `"CheckBox"` and `"ButtonRow"`. See examples on page 58.	Use to determine how the box is to be treated by the application. CheckBox is only supported for the three out-of-the-box check boxes on the Loan Search page.

The following attributes are for `Type="ButtonRow"`

Attribute	Required	Default	Explanation
`ResetButtonText`	No	`"Reset"`	Use to modify the text that appears on the [out-of-the-box] **Reset** button on search pages. Note that if there is no value for this attribute (in other words, `Reset=""`), this button will not appear on the search page.
`SubmitButtonText`	Yes	`"Submit"`	Use to modify the text that appears on the [out-of-the-box] **Submit Query** button on search pages.

The following attributes are for `Type="CheckBox"`

Attribute	Required	Default	Explanation
`chkID`	Yes	None	Currently only implemented for the three check boxes on the Loan Search page. This value of `chkID` is not configurable.
`ShowCheckBox`	No	`"true"`	Set to `"true"` to display check boxes on the Loan Search page or to "false" to hide them.

Example of the `FieldList` attribute with multiple fields for the Borrower Search page

```
<inmg:QueryScreen Name="borrower_search" Title="Borrower Search"
       BrowseButtonText="Browse" BooleanAnd="AND" BooleanOr="OR"
       BooleanNot="NOT" >
    <QueryField Type="QueryField" Display="Any Word"
       CssClass="SearchTextInput" LabelCssClass="SearchLabel"
       ShowBrowse="false" ShowBoolean="true" FieldList="BorrNumber
       | BorrName | BorrDepartment | BorrTelephone | BorrFax | BorrEmail
       | BorrAddress | BorrMailStop | BorrCity | BorrState | BorrPostalCode
       | BorrCountry"/>
    <QueryField Type="QueryField" CssClass="SearchTextInput"
       LabelCssClass="SearchLabel" ShowBrowse="true"
       ShowBoolean="true" FieldList="BorrName"/>
    ...
</inmg:QueryScreen>
```

Example of the `FieldList` attribute with secondary textbase fields for the Loan Search page

```
<inmg:QueryScreen Name="loans_search" Title="Loan Search"
       BrowseButtonText="Browse" BooleanAnd="AND" BooleanOr="OR"
       BooleanNot="NOT">
    <QueryField Type="QueryField" Display="Borrower"
       CssClass="SearchTextInput" LabelCssClass="SearchLabel"
       ShowBrowse="true" ShowBoolean="true"
       FieldList="BorrName.LoanBorrID | BorrNumber.LoanBorrID"/>
    ...
</inmg:QueryScreen>
```

Example of the `Type` attribute with values other than `QueryField` for the Loan Search page

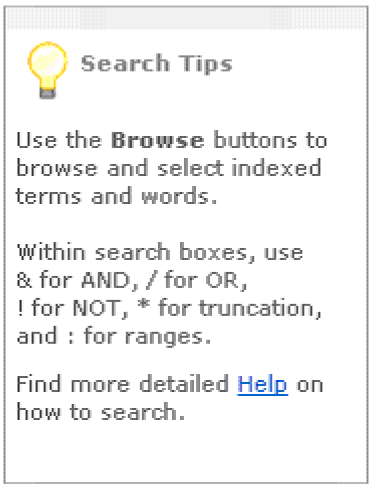
```
<inmg:QueryScreen Name="loans_search" Title="Loan Search"
       BrowseButtonText="Browse" BooleanAnd="AND" BooleanOr="OR"
       BooleanNot="NOT">
    <QueryField Type="QueryField" Display="Borrower"
       CssClass="SearchTextInput" LabelCssClass="SearchLabel"
       ShowBrowse="true" ShowBoolean="true"
       FieldList="BorrName.LoanBorrID | BorrNumber.LoanBorrID"/>
...
    <QueryField Type="CheckBox" Display="Overdue loans only" CssClass="
       SearchCheckBox" chkID="Overdue" ShowCheckBox="true" />
    <QueryField Type="ButtonRow" SubmitButtonText="Submit Query"
       ResetButtonText="Reset" CssClass="submit_button"/>
</inmg:QueryScreen>
```

Configuring the Search Tips Box

You can use the MyQueries.CONFIG file to configure the Search Tips box that appears on each search page (including making it not appear). You may find this feature useful if you have set up the application to provide more than one search page per textbase (see page 51) so that you can provide your own tips for your search page(s).

The following illustration shows the default Search Tips box. Out-of-the-box, *Genie* comes with the Search Tips box displayed on all the search pages provided by the application. To configure the Search Tips box for a search page, either provided by the application or created by you, you must add a `SearchTips` element within the `QueryScreen` element for that search page.

Out-of-the-box, the MyQueries.CONFIG file includes a `SearchTips` element section in the `catalog_search` section as an example. The attributes for this element are the same as configured for the default Search Tips box.



To configure the Search Tips box

1. Open the MyQueries.CONFIG file.
2. Go to the `QueryScreen` element section for the search page for which you want to configure the Search Tips box.
3. Before the `</inmg:QueryScreen>` line, enter a `SearchTips` element. The following example shows this added element to the search page for the `catalog_search` section (which is for the CATALOG textbase):

```
<inmg:QueryScreen Name="catalog_search"   Title="Catalog Search"...>
   <QueryField Type="QueryField"  Display="Any Word"...>
   <QueryField Type="QueryField"  Display="Title"...>
   <QueryField Type="QueryField"  Display="Author"...>
   <QueryField Type="QueryField"  Display="Subject"...>
   ...
   <QueryField Type="ButtonRow"  SubmitButtonText="  Submit Query "...>
      <SearchTips ShowTips="true"  ShowTitle="true"  ShowIcon="true"
      Title="Search Tips"  ImageURL="Images/xp_bulb.gif"  UseDefaultTips="false"
      ShowHelpLink="true"  TipContent="Use only UPPERCASE letters in the Author
      query box."/>
</inmg:QueryScreen>
```

4. Save the configuration file and reset *Genie*.

The `SearchTips` Element

In the MyQueries.CONFIG file, the `SearchTips` element contains its configuration settings in the format string (list of attributes). The following table lists the supported attributes for this element.

Attribute	Required	Default	Explanation
`ImageURL`	No	`"Images/ xp_bulb.gif"`	Provides the location for the image to be used as an icon on the Search Tips box. Use in conjunction with the `ShowIcon` attribute. The default, `xp_bulb.GIF`, is located in the Images subfolder of the main *Genie* installation folder. You can enter any path that the application can access.
`ShowHelpLink`	No	`"true"`	Sets whether or not to show the default Help link provided with the application, which is shown in the illustration on page 59.
`ShowIcon`	No	`"true"`	Sets whether or not to show an icon before the title of the Search Tips box. When set to `"true"`, you must provide the path to an image file in the `ImageURL` attribute. Set to "false" if you do not want to show an icon.
`ShowTips`	No	`"true"`	Sets whether or not to show the Search Tips box. Set to "false" if you do not want the Search Tips box to appear on a search page.
`ShowTitle`	No	`"true"`	Sets whether or not to show a title for the Search Tips box. When set to `"true"`, provide the title you want to appear in the `Title` attribute. Set to "false" if you do not want a title to appear.
`Title`	No	`"Search Tips"`	Sets the title for the Search Tips box. Use in conjunction with the `ShowTitle` attribute.
`TipContent`	No	See illustration on page 59.	Enter your own tips text. Use in conjunction with the `UseDefaultTips` attribute when appropriate.
`UseDefaultTips`	No	`"true"`	Sets whether or not to use the default tips text, which is shown in the illustration on page 59. Use in conjunction with the `TipContent` attribute if you set this attribute to "false".

MyReports.CONFIG File

The MyReports.CONFIG file defines all the reports provided for the application. You can also use it to create and add new reports. Reports are grouped by textbase. The groupings are listed in textbase-alphabetical order in the file (so that the reports for the BORROWER textbase are listed first and the ones for the SUPPLIER textbase are listed last).

You can make changes to the following levels in the MyReports.CONFIG file:

- **Textbase.** At this level (`ReportList` element) you can control what report choices appear in the drop-down list for a textbase. This section is at the end of the configuration file, after the SupplierFullReport. See page 62.
- **Report.** At this level (`Report` element) you can create new reports and define their report configurations. See page 64.
- **Field.** At this level (`Field` element) you can specify the textbase fields included in a report. See page 69.
- **EmailInfo**. At this level (`EmailInfo` element) you can specify parameters to use when sending e-mail that uses this report to create the body. See page 81.
- **ActionInfo**. At this level (`ActionInfo` element) you can specify parameters to use to update records that have been processed (for example, sent as e-mail) . See page 82.

To create a new report, we suggest you copy a report already defined in the MyReports.CONFIG file that best resembles the report you want to make and use it as a template for your new report. For your new report:

1. Change the `Name` attribute for the `Report` element (see page 66), which is required and must be unique for every report, and, if needed, its report configuration. Typically, this name does not appear in the user interface. The name should avoid characters that need to be escaped in XML, including ampersand, angle brackets, and extended characters. It may be useful to limit the use of all punctuation in form names. Case is also significant.
2. Add, delete, and/or change the `Field` elements (see page 69) as needed to set up the textbase fields. For a list of the textbase fields available for a textbase, open the MyGenieFieldMaps.CONFIG file, which is located in the ConfigFiles subfolder of the main *Genie* installation folder (for example, C:\Program Files\Inmagic\Genie\ConfigFiles).
3. Use the `ReportList` element (see page 62) to add the new report to the drop-down list for a textbase so that the report can be used.

Print-Friendly Reports

Reports can be set up to be print-friendly. Such reports omit the menus, page headings, navigation controls, etc. They contain a minimal set of controls, such as a drop-down list for selecting a different report. These controls do not appear on the report when it is printed.

Print-friendly reports use the batch_print.aspx page. Labels defined in MyLabels.config are also print-friendly.

Print-friendly reports depend upon the presence of an ActiveX control on the computer doing the printing. This control, ScriptX from MeadCo, is shipped with Genie, and you may be asked to install it the first time you initiate a print-friendly report on a computer. Print-friendly reports require an Internet Explorer browser.

Making Textbase-level Changes for Reports

At the very end of the MyReports.CONFIG file is the section that gives you control of reports at the textbase level. In this section, the `ReportList` element specifies that there are reports associated with a textbase. The `Report` element within the `ReportList` element specifies a report. See the example below. By default, each textbase has a Summary report (the multiple-record Search Results page that appears after you do a search) and a Full Display report (the report that appears when you click the **Full Display** link on a Search Results page, which shows one record at a time with box labels on the left and field contents on the right). If a textbase is listed in this section of the configuration file, there are more report choices available than the Summary and Full Display pages. When this is the case, after a search is performed, the Summary report will display a drop-down list that allows access to the other reports. The default Summary report will be the first report listed in the drop-down list.

Note that there are special virtual textbases for the OPAC (called Opac), for Items search exposed in the OPAC (called OpacItems), and for Serials Claim-related reports (called SerialsClaim). Functionally, when a user searches your OPAC, the search is of the CATALOG textbase, and when library staff do a search to claim missing issues (choose **Serials>Claim Missing Issues**) the search is of the SERIALS textbase. The `<inmg:ReportList TextbaseName="Catalog">` element is for the search results page of your Catalog records for your **library staff** and the `<inmg:ReportList TextbaseName="Serials">` element is for Serial records. To make it possible to show different reports in the drop-down list for the search results page for your OPAC and for the search results page for when library staff claim missing issues, use `<inmg:ReportList TextbaseName="Opac">` and `<inmg:ReportList TextbaseName="SerialsClaim">` (see page 63), respectively.

Example of reports for the LOANS textbase

```
<inmg:ReportList TextbaseName="Loans">
<Report Name="LoansByTitle" Display="Loans By Title"/>
<Report Name="LoansByBorrower" Display="Loans By Borrower"/>
<Report Name="LoansRenewal" Display="Loans Renewal"/>
<Report Name="LoansByCallNumber" Display="Loans By Call Number"
    PageName="batch_print.aspx"/>
<Report Name="LoansStatsByTitle" Display="Loans Statistics By Title"
    PageName="batch_print.aspx"/>
<Report Name="LoansStatsByClass" Display="Loans Statistics By Class"
    PageName="batch_print.aspx"/>
<Report Name="LoansStatsByDepartment" Display="Loans Statistics By Department"
    PageName="batch_print.aspx"/>
...
</inmg:ReportList>
```

Example of reports for the LOANS textbase, with changes

In this example, a LoansByDueDate report has been added to the list and placed between the LoansByRenewal and LoansByCallNumber reports. Note that LoansByDueDate is not a report provided with the application. It must be created in the Reports section of the configuration file before it can be referred to here.

```
<inmg:ReportList TextbaseName="Loans">
<Report Name="LoansByTitle" Display="Loans By Title"/>
<Report Name="LoansByBorrower" Display="Loans By Borrower"/>
<Report Name="LoansRenewal" Display="Loans Renewal"/>
<Report Name="LoansByDueDate" Display="Loans By Due Date"/>
<Report Name="LoansByCallNumber" Display="Loans By Call Number"
    PageName="batch_print.aspx"/>
...
</inmg:ReportList>
```

Example of reports for the virtual textbase SerialsClaim

```
<!-- ** Note: Use special TextbaseName="SerialsClaim" to specify the reports
   listed in serial claim related reports -->
<inmg:ReportList TextbaseName="SerialsClaim">
<Report Name="SerialsClaim" Display="Serials Claim"/>
<Report Name="PrintSerialsClaim" Display="Print Serials Claim"
   PageName="batch_print.aspx"/>
</inmg:ReportList>
```

The following table lists the supported attributes for the `Report` element in the `ReportList` sections.

Attribute	Required	Explanation
`Display`	Yes	Provides the user-friendly name for a report to display in the drop-down list. Applies to Summary reports.
`Name`	Yes	Identifies the Report Name attribute. It should be the same as the Report Name attribute defined for a report in the report definition. Applies to Summary reports.
`PageName`	No	Identifies the name of the page that displays the report. It defaults to the TextbaseName_report.ASPX page (for example, catalog_report.ASPX). For printed reports, we recommend you direct the page to the batch_print.ASPX page (which uses the CSS classes in the Print.CSS file). The batch_print.ASPX page supports page breaks in reports. Applies to Summary reports.
`Type`	No	Specify whether the `Report Name` is a report or label.

Example of use of Type="Label"

```
<inmg:ReportList TextbaseName="Serials">
<Report Name="SerialsBrief" Display="Serials Brief"/>
<Report Name="SerialsCheckInOrSkip"  Display="Check in or skip serials"/>
<Report Name="SerialsRenewal" Display="Renewal List" PageName="batch_print.aspx"/>
<Report Name="Route1on8.5x11"  Display="Print Routing Notices"
   PageName="batch_print.aspx" />
<!-- Sample configuration for adding a Label to the report drop-down list -->
<Report Name="LabelRoute3on8.5x11" Display="Print Routing List on Label"
   Type="Label"/>
</inmg:ReportList>
```

Creating New Reports at the Report Level

You should make any changes through your MyReports.CONFIG file.

Every report provided with the *Genie* application is identified and defined in the MyReports.CONFIG file. You use the `Report` element to identify a report. When you create a new report, you must include a unique `Report` element. Within it, you can specify configuration information that applies to that report. After you create the `Report` element, you then must specify the textbase fields for the new report (see page 69).

Example of OrdersClaim report and its configuration information

```
<!-- OrdersClaimReport -->
<inmg:Report Name="OrdersClaim" CellPadding="0"
   GroupBy="SuppName.OrdSuppID" PageBreak="true"
   SortMethod="specify" SortBy="SuppName.OrdSuppID|OrdCatTitle">
   ...
</inmg:Report>
```

The following table lists the supported attributes for the `Report` element for the MyReports.CONFIG file. Note that links used to initiate an action (such as `AddAllToCart`) appear at both the top and bottom of the current page in the Summary display. In most cases other than actions involving the InfoCart, the action buttons are used with `%%CheckBox` elements, and act on the selected records.

Attribute	Required	Default	Explanation
`AddAllToCart`	No	`"false"`	When set `"true"`, displays an **Add all to Cart** link to the Catalog or OPAC Summary reports. When a user clicks on the link, the application adds all the Catalog records on the current page to the InfoCart. See also `RemoveAllFromCart` on page 67.
`AddButton`	No	`“false”`	On a serials report, places a button to add a borrower name to the routing list for selected subscriptions. Used with `%%CheckBox`.
`AddButtonText`	No	`None`	Text that appears for the `AddButton`.
`BatchDelete`	No	`"false"`	Adds a **Batch Delete** link to the Summary report, if this function is supported for the current textbase and role. See page 26. Used with `%%CheckBox`.
`BatchModify`	No	`"false"`	Adds a **Batch Modify** link to the Summary report, if this function is supported for the current textbase and role. See page 26. Used with `%%CheckBox`.
`CellPadding`	No	5 pixels	Acts just like the `CellPadding` attribute for HTML where it sets the amount of space around all four sides of the record content. Applies to Summary reports. ***Tip!*** For reports that use the `GroupBy` attribute (see below in this table), we recommend you use `CellPadding="0"`.
`CheckInSerials`	No	`"false"`	Displays a **Check in** link on the top and bottom of each page of the Summary report for Serials. Used with `%%CheckBox`. See also `SkipMissingIssues` on page 67.
`ClearAll`	No	`"false"`	Displays a **Clear all** link on the Summary report. The **Clear all** link works with `%%CheckBox` (see page 77) to clear all the check boxes on the current page. See also `SelectAll` on page 67.

Continued…

Attribute	Required	Default	Explanation
`DefaultClass`	No	None	Default CssClass to use in the report if not otherwise specified in a Field element.
`EmailClaim`	No	None	On a serials, orders, or ILL report, places an “E-mail” link that causes a claim report to be sent via e-mail. The value is the name of the report that will generate the e-mail. Used with `%%CheckBox`.
`EmailOverdues`	No	None	On a loans report. places a link that causes an overdue notice to be sent via e-mail. The value is the name of the report that will generate the e-mail. Used with `%%CheckBox`.
`EmailRoute`	No	None	On a serials report, places an “E-mail” link that causes a routing report to be sent via e-mail. The value is the name of the report that will generate the e-mail. Used with `%%CheckBox`.
`Form`	No	None	**In *Genie* v3.00 and later, this attribute is deprecated and not recommended**. Sort parameters are specified in MyReports.CONFIG, and field lists are assembled from the content of the report definition. If used, the `Form` attribute overrides the sort parameters and fields in the report definition. The `Form` attribute names a form designed within the textbase. That form definition lists the fields to be returned in the XML result, and controls the order in which the records are returned.
`GroupBy`	No	None	Allows you to organize a report by group. The field listed should usually be the the first field listed for the SortBy parameter. Use `GroupBy="%%Null"` if you want all the records treated as a single group. When `GroupBy="%%Null"` and `PageBreak="true"` (see page 66), a page break is inserted after every record. Applies to Summary reports. Note that if the field you are referring to is a field in a secondary textbase, then you must specify it in the format SecondaryFieldName .PrimaryLinkFieldName. For example, the OrdersClaim report for the ORDERS textbase contains the supplier name, which resides in the SUPPLIER textbase. In the OrdersClaim report for the `GroupBy` attribute, this reference is entered as `GroupBy="SuppName.OrdSuppID"`, where `SuppName` refers to the *SuppName* field in the secondary textbase (SUPPLIER), and `OrdSuppID` the Link field in the primary textbase (ORDERS).
`HidePanel`	No	`"false"`	Hides the general panel at the top of the page containing the header and history links.
`ItemNumberWidth`	No	0 or 40	Default depends on whether `ShowItemNumber` (below in this table) attribute is `"true"`. Applies to Summary reports. ***Tip!*** You can use this attribute to indent the report when there is no item number.

Continued…

Attribute	Required	Default	Explanation
`MarkAsRouted`	No	`"false"`	Used for the SERIALS textbase. If set to `"true"`, a **Mark As Routed** button will be displayed on the batch print page. After printing, you would click the **Mark As Routed** button to batch clear the route dates. See also the `MarkAsRouted` attribute for the MyLabels.CONFIG file on page 40. ***Note:*** This only works for the batch print page and the label print page. It does not work for regular report pages.
`MaxItems`	No	100	Used with `ShowItemInfo="true"`or the <ItemInfo> section, specifies the maximum number of items to display on the Catalog Full Display page.
`Name`	Yes		The name of the report, called by the *Genie* report display page. Applies to both Summary and Full Display reports.
`OneRecordPerGroup`	No	`"false"`	Used in statistical reports to display one record per group. Must be used together with the `GroupBy` attribute (see above in this table). When used with batch_print.ASPX to generate a print-friendly report, the `PageBreak` attribute (next row in this table) should be set to false. (Note that the batch_print.ASPX page uses the CSS classes in the Print.CSS file.) Applies to Summary reports.
`PageBreak`	No	`"false"`	Used by the batch_print.ASPX page. When set to `"true"`, a page break is inserted after each group if the `GroupBy` attribute (see above in this table) is set. Otherwise, a page break is inserted after each record. When false, no page break is inserted.
`PageSize`	No	`"20"`	Sets the number of records per page. Use `PageSize="0"` to show all records on one page. Applies to Summary reports.
`PercentIncrease`	No	None	Used with the `ProjectedTotalCost` function (see page 79) for Summary reports to specify the percentage of the increase. You can specify fractional (for example, .5) and negative (for example, -10) values.
`PrintFooter`	No	None	Used with the batch_print.ASPX page to specify the information to appear in the bottom margin on every page of the printout. See the Codes for Headers and Footers table on page 80. Applies to Summary reports and when you use batch_print.ASPX.
`PrintHeader`	No	None	Used with the batch_print.ASPX page to specify the information to appear in the top margin on every page of the printout. See the Codes for Headers and Footers table on page 80.

Continued...

Attribute	Required	Default	Explanation
`PrintOverdues`	No	None	On a loans report, places a link that will generate a print-ready overdue notice report. The value is the name of the report. Used with `%%CheckBox`.
`RemoveAllFromCart`	No	`"false"`	Displays a **Remove all from Cart** link. When a user clicks the link, the application removes all the Catalog records on the current page from the InfoCart. See `AddAllToCart` on page 64.
`RemoveButton`	No	`“false”`	On a serials report, places a button to remove a borrower name from the routing list for selected subscriptions. Used with `%%CheckBox`.
`RemoveButtonText`	No	`None`	Text that appears for the RemoveButton.
`RenewSelectedItems`	No	`"false"`	Displays a **Renew selected items** link. The **Renew selected items** link works with `%%CheckBox` (see page 77) to execute the loans renewal or serials subscription renewal operation.
`SelectAll`	No	`"false"`	Displays a **Select all** link on the Summary report. The **Select all** link works with `%%CheckBox` (see page 77) to select all the records on the page. See also `ClearAll` on page 64.
`ShowEditItemLink`	No	`"false"`	Displays an **Edit Items** link on the Catalog Full Display page if the Catalog record has any associated Item records.
`ShowEditReserveLink`	No	`"false"`	Displays an **Edit the reserve list** link on the Catalog Full Display page.
`SkipMissingIssues`	No	`"false"`	Displays a **Skip missing issues** link on the Summary report for Serials. Used with `%%CheckBox`. See also `CheckInSerials` on page 64.
`ShowBorrowerInfo`	No	`"false"`	Displays borrower information along with the item information on the Catalog or OPAC Full Display page. For example, if you set `ShowItemInfo="true"` but `ShowBorrowerInfo="false"`, end users will see the Item Information (bar code and copy information), and whether the item is on loan, and when it is due, but not who has it out. Applies to Full Display reports.

Continued...

Attribute	**Required**	**Default**	**Explanation**
`ShowItemInfo`	No	`"false"`	Displays data from the ITEMS textbase in the Catalog and OPAC Full Display reports. For example, `ShowItemInfo="false"` means no item information appears at all. The Item Information section is where active loans are shown. For example, if you set `ShowItemInfo="true"` but `ShowBorrowerInfo="false"`, end users will see the Item Information (bar code and copy information), and whether the item is on loan, and when it is due, but not who has it out. Applies to Full Display reports.
`ShowItemNumber`	No	`"false"`	Numbers the records in the report. ***Note:*** If this attribute is set to "false" in the MyReports.CONFIG file for any Catalog or OPAC Summary report, the red check mark (which acts as a visual signal to indicate that an item has been added to the InfoCart) will not appear next to a title after the item has been added to the InfoCart. Note though that the link text for the InfoCart will change to **Remove from Cart** (instead of **Add to Cart**) once an item has been added to the InfoCart, so the text for the InfoCart link also acts a visual cue.
`ShowManageRouteListLink`	No	`“false”`	On a borrower full report, provides a link to support adding/removing a borrower name on routing lists for selected subscriptions.
`ShowReserves`	No	`"false"`	Displays the information from the RESERVES textbase in the Catalog and OPAC Full Display reports.
`SortBy`	No	None	Specify the fields by which to sort the report. Multiple fields are separated by ‘|’. If multiple fields are specified, the report is sorted by the first field listed, then subsorted by the next one, etc., up to a maximum of 5 fields total. Example: `SortBy="BorrDepartment|BorrName"`
`SortMethod`	No	`"textbase"`	Specify the sort method. Supported methods are: **specify**, **relevance**, **unsorted**, and **textbase**. Use **specify** if you are specifying the fields using SortBy. Use **textbase** to use the textbase default sort. Use **unsorted** to return the records in the order in which they were added to the textbase. Use **relevance** to return records ordered by relevance to the query performed, if the query involved any word searches.

Making Changes for Reports at the Field Level

You should make any changes through your MyReports.CONFIG file.

Every report provided with the *Genie* application is identified and defined in the MyReports.CONFIG file. This section covers the `Field` element, which you use to define each textbase field for a report. (The previous section covers the `Report` element, which you use to identify a report.) Note that you can specify functions with or without textbase fields. You can use the `Field` element to add layout and/or formatting information (for example, including a data table), text and/or HTML tags that are not related to a textbase field (for example, the content of a letter to a borrower with materials that are overdue), and action links (for example, the **Edit** link).

Important! Keep the following in mind:

- The textbase fields must have the `Name` attribute, which should be the same as the `AppName` attribute specified in the MyGenieFieldMaps.CONFIG file.
- A function can work with a textbase field. For example, the `LatestDate` function (see page 79) will work with the *LoanDateLoaned* field in the LoansStatsByTitle report.
- Some functions (see page 79) do not require a field name because they work with hardcoded fields. For example, the `TotalCost` function is hardcoded to the *OrdListPrice* and *OrdNumberOfCopies* fields. The `GrandTotalLoans`, `ProjectedGrandTotalCost`, and `GrandTotalCost` functions are always used with the `GroupFooter` attribute (see page 72), and they work with the corresponding `TotalCost`, `ProjectedTotalCost`, and `TotalLoans` functions.
- For layout and/or formatting, text and/or HTML tags that are not related to a textbase field, and action links, you should leave the `Name` element empty and use the predefined special tags in the element's inner text (see page 77).
- Each field listed in the MyReports.CONFIG file will show for every record unless you set `GroupHeader="true"` or `GroupFooter="true"`. In that case, they only appear in the first record or the last record.

Example of field configuration from OrdersClaim report

```
<!-- OrdersClaimReport -->
<inmg:Report Name="OrdersClaim" CellPadding="0"
   SortBy="SuppName.OrdSuppID|OrdCatTitle" GroupBy="SuppName.OrdSuppID"
   PageBreak="true">
   ...
   <Field ShowHeader="true" Width="620">%%StartTable</Field>
   <Field ColumnHeader="Copies&lt;br&gt;&lt;br&gt;">%%StartColumn</Field>
   <Field Name="OrdNumberOfCopies" Header=" "/>
   <Field>%%EndColumn</Field>
   <Field Width="290" ColumnHeader="Title">%%StartColumn</Field>
   <Field Name="OrdCatTitle"/>
   <Field>%%EndColumn</Field>
   <Field width="15">%%StartColumn</Field>
   <Field>%%EndColumn</Field>
   <Field Width="110" ColumnHeader="Order Date">%%StartColumn</Field>
   <Field Name="OrdDateOrdered"/>
   <Field>%%EndColumn</Field>
   <Field Width="100" ColumnHeader="Order Number">%%StartColumn</Field>
   <Field Name="OrdOrderNumber"/>
   <Field>%%EndColumn</Field>
   <Field>%%EndTable</Field>
   <Field CellSpacing="0" CellPadding="0">%%StartTable</Field>
   <Field Width="602">%%StartColumn</Field>
   <Field Display="&lt;br&gt;&lt;br&gt;Please let us know when we may
   expect delivery. Thank you. &lt;br&gt;&lt;br&gt;&lt;br&gt;&lt;br&gt;"
   GroupFooter="true">%%Label</Field>
   <Field>%%EndColumn</Field>
   <Field>%%EndTable</Field>
</inmg:Report>
```

The following table lists the supported attributes for the `Field` element for the MyReports.CONFIG file.

Attribute	Required	Default	Explanation
`AddField`	No	None	Used in the Serials Routing List reports to append a Borrower field value after the corresponding entry in the SerRouteList field.
`AddFieldSeparator`	No	None	Used with AddField to specify the text to appear between the SerRouteList entry and the Borrower field. For example, an ellipsis.
`BindingField`	No	None	Used for `%%CheckBox` special tag (see page 77). If no binding field is specified, the check box will be displayed for each record unconditionally. If a binding field is specified, the check box will be displayed only if the specified field is not empty. Applied to Summary reports.
`CssClass`	No	For Summary report: `SearchResultsText` For Full Display report: `display_text`	CSS class, defined in Genie.CSS applied to field contents, headers, footers, separators, and the `%%Label` special tag (see table on page 77). The Genie.CSS file resides in the main *Genie* installation folder. Applies to Summary and Full Display reports.
`Display`	No	Mapped to the `BoxLabel` attribute in the MyGenieFieldMaps .CONFIG file (see page 32) for Full Display reports. None for Summary reports.	Used in Full Display reports; it overrides the BoxLabel specified in the MyGenieFieldMaps.CONFIG file. Used in Summary and Full Display reports to display information provided by the `%%Label` special tag (see table on page 77).
`DisplayCss`	No	`display_label`	CSS class, defined in Genie.CSS, used for displaying box labels in a Full Display report. The Genie.CSS file resides in the main *Genie* installation folder. The `DisplayCss` with the largest width value controls the width of all the box labels on the Full Display report.
`Footer`	No	None	Used to add extra characters and HTML tags after the last entry of field content. Does not appear if the field is empty. Applies to Summary and Full Display reports. ***Note:*** If the end of the field and the beginning of the footer use the same punctuation, the one from the footer is omitted (to prevent double punctuation, such as double periods).

Continued…

Attribute	Required	Default	Explanation
`GroupFooter`	No	Regular display	Used to display data only at the end of a group of data. Must be used together with the `GroupBy` attribute (see page 65). Applies to Summary reports.
`GroupHeader`	No	Regular display	Used to display data only at the beginning of a group of data. Must be used together with the `GroupBy` attribute (see page 65). You may want to use `GroupHeader="true"` so that data appears only in the first record. Applies to Summary reports.
`Header`	No	None	Used to add extra characters and HTML tags before the first entry of field content. Does not appear if the field is empty. Applies to both Summary and Full Display reports.
`ImageHeight`	No	Original height	Used with `TreatAs="Image"` or `TreatAs="ToCImage"` only, to set display height (in pixels) in report.
`ImageWidth`	No	Original width	Used with `TreatAs="Image"` or `TreatAs="ToCImage"` only, to set display width (in pixels) in report. ***Important!*** If ImageHeight and ImageWidth are both set, the image will be resized to fit the specified dimensions. To preserve the aspect ratio of the original image, specify one of these parameters, not both.
`Name`	Yes, except for the special tags (see page 77)		Should be the same as the `AppName` attribute (see page 32) defined in the MyGenieFieldMaps.CONFIG file. Note that if the field you are referring to is a field in a secondary textbase, then you must specify it in the format SecondaryFieldName .PrimaryLinkFieldName. For example, the LoansByTitle report for the LOANS textbase contains the borrower name, from the BORROWER textbase. In the LoansByTitle report, the `Name` attribute is entered as `<Field Name="BorrName.LoanBorrID" />`, where `BorrName` refers to the *BorrName* field in the secondary textbase (BORROWER), and *LoanBorrID* the Link field in the primary textbase (LOANS). Applies to both Summary and Full Display reports.
`NoContentHeader`	No	None	Used to specify text that should appear when the content field is empty (for example, "permanent loan"). Applies to both Summary and Full Display reports.
`OneRecordPerEntry`	No	`"false"`	Used with fields that have multiple entries. If set to `"true"`, each entry will be treated as a separate event to be processed if it is selected. (Used in SerialsCheckInOrSkip report to apply a check box to each *SerDatesDue* field.) Applies to Summary reports.

Continued...

Attribute	Required	Default	Explanation
`Reverse`	No	`"false"`	Applies only to fields included in the `SortBy` attribute on the Report element. Set `Reverse="true"` to sort this field in reverse order, for example, most-recent-date first.
`Separator`	No	`"; "`	Used to add extra characters and HTML tags between multiple entries in the field. Applies to both Summary and Full Display reports.
`SortType`	No	`"sort"`	Applies only to extra fields included in the `SortBy` attribute on the Report element. Possible values are **sort** and **alternate**. Use **sort** to subsort by the specified field. Use **alternate** if the specified field is intended to be a substitute for the primary (first-listed) sort field if that field is empty.
`Style`	No	None	HTML style applied to field contents, headers, footers, separators, and the `%%Label` special tag (see table on page 77). Applies to both Summary and Full Display reports.
`TreatAs`	No	None	Possible values, which can be used in both Summary and Full Display: `"URL"`, `"Email"`, and `"SeeAlso"` will construct URL (including file URLs), e-mail, and See Also search links accordingly. `TreatAs="SeeAlso"` will only work as desired if the field has a Term index. `"FileLink"` will produce links to documents in the folder specified by the `DocumentFolder` element in MyGenie.CONFIG. `"Image"` and `"ImageLink"` will produce inline images or links to images in the folder specified by the `ImageFolder` element in MyGenie.CONFIG. `"ToCImage"`, `"ToCImageLink"`, and `"ToCIcon"` will produce inline images or links to table-of-contents images referenced from the Serials or Items textbases. They are located in the folder specified by the `ToCImageFolder` parameter in MyGenie.CONFIG. ***Note:*** Only `ToCImageLink` should be used in reports used for the body of e-mail messages. `"WordList"` will display each word of the Catalog Call Number field on a separate line. Use a non-breaking space character (Alt+0160) to keep two words (for example, "Vol. 1") together. Note that if the Catalog Call Number field has more than one entry, only the first call number will be used and all other entries will be ignored on Full Display reports.

Continued…

`UpperCase`	No	`"false"`	Changes field content to uppercase.
`UseFunction`	No	Regular text	For a list of the supported functions, see "Supported Functions" on page 79.
`UseTemplate`	No	None	A string containing within it a '%s' serving as a placeholder for the field content. This a way of supporting text that will appear before and/or after the field content without the insertion of <span> tags in a summary display. A typical use would be to create a link to a book cover on the web. See the note below.
`Visible`	No	`"true"`	Use `Visible="false"` in the special situation where the application needs information to determine whether the record is of a particular type but you do not need the field to appear in the report. For example, the *OrdSerFrequency* field in the OrdersBrief report is needed in the report definition to determine if the record is a serial- or item-related Order record so that the application can display the correct Add/Edit Order page, but the field itself does not need to appear in the report.

Important Note Regarding UseTemplate: This would be a sample report Field element to display the book cover from the URL at openlibrary.org. Inmagic does not endorse any specific book cover provider, and warns the user to check a site's terms and conditions regarding use.

```
<Field Name="CatISBN"
UseTemplate=" cover:&lt;br /&gt; &lt;img src='http://covers.openlibrary.org/b/isbn/%s-
S.jpg' /&gt;" />
```

The following attributes are related to tables in Summary reports

Attribute	Required	Default	Explanation
`Align`	No	`"Left"`	Applies to the `%%StartColumn` special tag (see page 77) to specify the column text alignment. Values can be `"Left"`, `"Right"`, and `"Center"`.
`CellPadding`	No	1 pixel	Applies to the `%%StartTable` special tag (see page 77) to specify the `CellPadding` for the table. ***Tip!*** For reports that use the `GroupBy` attribute, we recommend you use `CellPadding="0"`.
`CellSpacing`	No	1 pixel	Applies to the `%%StartTable` special tag (see page 77) to specify the `CellSpacing` for the table. ***Tip!*** For reports that use the `GroupBy` attribute, we recommend you use `CellSpacing="0"`.
`ColumnHeader`	No	None	Applies to the `%%StartColumn` special tag (see page 77) to specify the header text.
`HeaderAlign`	No	`"Left"`	Applies to the `%%StartColumn` special tag (see page 77) to specify the header text alignment. Values can be `"Left"`, `"Right"`, and `"Center"`.

`HeaderStyle`	No	None	Applies to the `%%StartColumn` special tag (see page 77) to specify the column header HTML style.
`ShowHeader`	No	`"false"`	Applies to the `%%StartTable` special tag (see page 77) to indicate that the column header should be displayed.
`VAlign`	No	`"Middle"`	Applies to the `%%StartColumn` special tag (see page 77) to specify the column vertical alignment. Values can be `"Top"`, `"Bottom"`, and `"Middle"`.
`Width`	No	570 pixels (for `%%StartTable`)	Applies to the `%%StartTable` or `%%StartColumn` special tags (see page 77) to specify the width of the table or column.

Catalog, OPAC, and Borrower Full Display Pages

This section details the element blocks for the Catalog, OPAC, and Borrower Full Display reports.

The `ItemInfo`, `LoanInfo`, and `ReserveInfo` blocks let you configure the Item, Reserves, and Loan information on the Catalog and OPAC Full Display pages.

The `SerialInfo` block lets you configure the Catalog and OPAC Full Display reports to show the most recently received Serial issue. On the Borrower Full Display report, the `SerialInfo` block is used to show the serials routed to the borrower.

ItemInfo Block

If present, the `ItemInfo` block will overwrite the *Genie* default item information with customized item information.

Example

```
<ItemInfo Heading="Item Information" ShowItemNumber="true">
   <Field Name="ItemCopyInfo" Footer=": "/>
   <Field Name="ItemBarCode"/>
   <Field Name="ItemLocation" Header=" "/>
   <Field Name="ItemSubLocation" Header=", "/>
   <Field Name="ItemShelf" Header=", " Footer=" " NoContentHeader=" "/>
</ItemInfo>
```

LoanInfo Block

The `LoanInfo` block works together with the `ItemInfo` block to display loan information. If you are using the `ItemInfo` block to overwrite the default item information, you need to specify the `LoanInfo` block also. Otherwise, no loan information will be displayed.

Example

```
<LoanInfo>
   <Field Name="BorrName.LoanBorrID" Header=" - on loan to " Footer=", "
      NoContentHeader=" - on loan, "/>
   <Field Name="LoanDateDue" Header="due " NoContentHeader="permanent loan"/>
</LoanInfo>
```

Tip! `LoanInfo` relies on `ItemBarCode` to retrieve data. If you want `LoanInfo` to be displayed but not `ItemBarCode`, you can set the `ItemBarCode` field to `Visible="false"` as in the following example:

```
<ItemInfo Heading="Item Information" ShowItemNumber="true">
   <Field Name="ItemCopyInfo" Footer=": "/>
   <Field Name="ItemBarCode" Visible="false"/>
   <Field Name="ItemLocation" Header=" "/>
   <Field Name="ItemSubLocation" Header=", "/>
   <Field Name="ItemShelf" Header=", " Footer=" " NoContentHeader=" "/>
</ItemInfo>
```

ReserveInfo Block

If present, the `ReserveInfo` block will overwrite *Genie* default reserve information to specify customized reserve information.

Example

```
<ReserveInfo Heading="Reserves" ShowItemNumber="true">
   <Field Name="ResDateRequested" Footer=", "/>
   <Field Name="BorrName.ResBorrID" Header="by "/>
   <Field Name="ResLocation" Header=", at location "/>
   <Field Name="ResItemBC" Header=", for item "/>
</ReserveInfo>
```

SerialInfo Block

If present, the `SerialInfo` block will display serial information on Catalog, OPAC, or Borrower Full Display page.

Example

```
<SerialInfo Heading="Serials" ShowItemNumber="true"   ViewDetail="true">
   <Field Name="SerDatesRecd" UseFunction="FirstEntry"
      Header="Latest issue received: " Footer=". "
      NoContentHeader="No issues received. "/>
</SerialInfo>
```

The following table lists the attributes supported for `ItemInfo`, `ReserveInfo`, and `SerialInfo` elements. The fields included in these elements have the same attributes support as other fields in the regular Full Display report, except the **SeeAlso** link, which is not supported. There are no attributes for `LoanInfo`, which works in tandem with `ItemInfo`.

Attribute	Required	Default	Explanation
`CssClass`	No	`"display_head"`	CSS class, defined in Genie.CSS, applied to the `Heading` attribute. The Genie.CSS file resides in the main *Genie* installation folder.
`Heading`	Yes		Display heading for item information section.
`Style`	No	None	HTML style applied to the `Heading` attribute.
`ShowItemNumber`	No	`"true"`	Numbers the Items, Reserves, or Serials records.
`SortBy`	No	None	Identifies field(s) by which the Serial or Item records should be sorted. Separate multiple fields with | (vertical bar).The records will be sorted by the first field in the list, subsorted by the second, etc. Precede a field name with a hyphen (-) to specify a reverse-sort by that field.
`ViewDetail`	No	`"false"`	Applies to Serial information in a Catalog or OPAC report only. Provides a **View Detail** link to view the SerialsDetail report.

Special Tags

The following table lists predefined special tags that you can use in the report configuration. You enter predefined tags as a field element's inner text in the MyReports.CONFIG file.

Note: These special tags are only supported in the Summary report, except for `%%Label`, `%%CartAdd`, and `%%CartRemove`, which are supported in the Full Display reports also.

Tag	Explanation
`%%StartTable`	Used in tabular layout. Marks the beginning of an HTML table. Any fields inserted after this and before the `%%EndTable` tag will be displayed inside this table. A table must have at least one column. Otherwise, the program ignores the tag. See the example on page 78 showing the OrdersBrief report.
`%%EndTable`	Marks the end of a table. See the example on page 78 showing the OrdersBrief report.
`%%StartColumn`	Marks the beginning of a table column. Any fields inserted after this and before the `%%EndColumn` tag will be displayed inside the column. It must be within the `%%StartTable` and `%%EndTable` block. See the example on page 78 showing the OrdersBrief report.
`%%EndColumn`	Marks the end of a column. See the example on page 78 showing the OrdersBrief report.
`%%DetailButton`	Constructs a **Full Display** link. See the example on page 78 showing the OrdersBrief report.
`%%EditButton`	Constructs an **Edit** link. See the example on page 78 showing the OrdersBrief report.
`%%CheckBox`	Used for batch options to display a check box next to each record. See the example on page 78 showing the SerialsCheckInOrSkip report.
`%%CartAdd`	Constructs an **Add to Cart** link. See the example on page 78 showing the CatalogBrief report.
`%%CartRemove`	Constructs a **Remove from Cart** link. See the example on page 78 showing the CatalogBrief report.
`%%Label`	Displays any text and/or HTML tags that are not related to a textbase field. See the example on page 78 showing the SerialsCheckInOrSkip report.
`%%VerticalBar`	Displays a vertical bar (|). See the example on page 78 showing the OrdersBrief report.

Example of OrdersBrief report showing some special tags

```
<!-- OrdersBriefReport -->
<inmg:Report Name="OrdersBrief" GroupBy="OrdOrderNumber" SortBy="OrdOrderNumber">
   <Field Name="OrdOrderNumber" Header="" Style="font-weight:bold"
   GroupHeader="true"/>
   ...
   <Field>%%StartTable</Field>
   <Field Width="20">%%StartColumn</Field>
   <Field>%%EndColumn</Field>
   <Field>%%StartColumn</Field>
   <Field Name="OrdItemNumber"  Style="font-weight:bold" Footer=": " />
   <Field Name="OrdCatTitle"  Style="font-weight:bold"/>
   <Field Name="BorrName.OrdReqID" Header=" -- Requested by " />
   <Field Name="OrdStatus" Header=" (" Footer=")" />
   <Field Header="&lt;br/&gt;&lt;br/&gt;">%%EditButton</Field>
   <Field>%%VerticalBar</Field>
   <Field Footer="&lt;br/&gt;">%%DetailButton</Field>
   <Field>%%EndColumn</Field>
   <Field>%%EndTable</Field>
</inmg:Report>
```

Example of CatalogBrief report showing some special tags

```
<inmg:Report Name="CatalogBrief" ShowItemNumber="true"
   SortBy="CatTitle|CatDatePublished"
   AddAllToCart="true" RemoveAllFromCart="true">
   <Field Name="CatTitle" Header="" Style="font-weight:bold" />
   <Field Name="CatSubtitle" Header=": " />
   <Field Name="CatAuthor" Header=" / " Separator="; " Footer="." />
   <Field Name="CatCorporateAuthor" Header=" / " Separator="; " Footer="." />
   <Field Name="CatPlace" Header=" -- " />
   <Field Name="CatPublisher" Header=": " />
   <Field Name="CatDatePublished" Header=", " Footer="." />
   <Field Name="CatRecordType" Header=" &lt;i&gt;(" Footer=")&lt;/i&gt; " />
   <Field Name="CatCallNumber" Header=" " />
   <Field Header="&lt;br/&gt;&lt;br/&gt;">%%EditButton</Field>
   <Field>%%VerticalBar</Field>
   <Field>%%DetailButton</Field>
   <Field>%%VerticalBar</Field>
   <Field>%%CartAdd</Field>
   <Field>%%CartRemove</Field>
   <Field Display="&lt;br/&gt;&lt;br/&gt;">%%Label</Field>
</inmg:Report>
```

Example of SerialsCheckInOrSkip report showing some special tags

```
<!-- SerialsCheckInOrSkipReport -->
<inmg:Report Name="SerialsCheckInOrSkip" ShowItemNumber="false" PageSize="20"
   GroupBy="SerID" CheckInSerials="true" SkipMissingIssues="true" SelectAll="true"
   ClearAll="true">
   <Field Name="SerTitle" GroupHeader="true" Style="font-weight:bold" />
   <Field Name="SerCopy"  GroupHeader="true" Header=" Copy:"/>
   <Field Display="&lt;br/&gt;&lt;br/&gt;" GroupHeader="true" >%%Label</Field>
   <Field Width="620" CellSpacing="0" CellPadding="0">%%StartTable</Field>
   <Field Align="Left" Width="25" VAlign="Center">%%StartColumn</Field>
   <Field BindingField="SerDatesDue">%%CheckBox</Field>
   <Field>%%EndColumn</Field>
   <Field Align="Left" VAlign="Center">%%StartColumn</Field>
   <Field Name="SerDatesDue" Header="" OneRecordPerEntry="true"
   UseFunction="NumberOfEntries(5)" />
   <Field>%%EndColumn</Field>
   <Field>%%EndTable</Field>
</inmg:Report>
```

Supported Functions

The following table lists the functions that you can use in the report configuration. These functions are only supported in Summary reports, except where noted.

Function	Explanation
AverageLoanDays	Calculates the average loan days of a group. (*LoanDateReturned* - *LoanDateLoaned*) / *#RecordsInGroup*. Only supported in the LOANS textbase. Must be used with the GroupBy attribute (see page 65) and OneRecordPerGroup="true" (see page 66).
CountDistinct	Counts the number of distinct items in a field in a group of records.
CountEntries	Returns the number of entries. Can also be used in Full Display reports.
Date	Displays today's date. Can also be used in Full Display reports.
FirstEntry	Returns the first entry. Can also be used in Full Display reports.
From	Displays "From" information. Uses information supplied by the Set Up E-mail Information page. Can also be used in Full Display reports.
GrandAverageLoanDays	Calculates a grand average of days on loan across all groups (Total days / #Records). Only supported in the LOANS textbase. Use with the OneRecordPerGroup attribute (see page 66) and the GroupBy attribute (see page 65).
GrandTotalCost	Displays the total of the TotalCost column. Only supported in the ORDERS textbase.
GrandTotalLoans	Calculates a grand total number of loans across all groups. Only supported in the LOANS textbase. Use with the OneRecordPerGroup attribute (see page 66) and the GroupBy attribute (see page 65).
IsActiveReserve	Displays an asterisk (*) if the item on loan has an active reserve against it. Use for an overdue loan report. The report must include the *LoanItemBC* and *LoanCatID* fields. Hide these fields (for example, <Field Name="LoanItemBC" Visible="false"/>) if you do not want them to appear in the report.
LatestDate	Compares dates and returns the most recent date.
NumberOfEntries(x)	Used with fields that have multiple entries. Specify *x* number of entries to be displayed. By default, all entries will be displayed. Can also be used in Full Display reports.
NumberOfWords(x)	Specify *x* number of words to be displayed. Only the first entry is taken into consideration. Can also be used in Full Display reports.
ProjectedGrandTotalCost	Total of the ProjectedTotalCost column. Only supported in the ORDERS textbase.
ProjectedTotalCost	Calculates projected total cost based on the PercentIncrease attribute (see page 66) of the report. TotalCost x (1 + PercentIncrease/100). Only supported in the ORDERS textbase.
TotalCost	Calculates total cost of an order (*OrdListPrice* x *OrdNumberOfCopies*). Only supported in the ORDERS textbase.
TotalLoans	Displays total loans in a group. Only supported in the LOANS textbase. Used with the GroupBy attribute (see page 65) and the OneRecordPerGroup="true" attribute (see page 66).

Codes for Headers and Footers

To print specific information as part of the page header or footer, you can include the following codes as part of the `PrintHeader` or `PrintFooter` attribute for the Report element. (See page 66.) Note that these are Microsoft Internet Explorer parameters that are provided here for your convenience.

Use…	When you want to show…
`&w`	the window title
`&u`	the page address (URL)
`&d`	the date in the short format as specified by Regional Settings in the Control Panel
`&D`	the date in the long format as specified by Regional Settings in the Control Panel
`&t`	the time in the format specified by Regional Settings in the Control Panel
`&T`	the time in 24-hour format
`&p`	the current page number
`&P`	the total number of pages
`&&`	a single ampersand (&)
`&b`	the text immediately following this code right-justified
`&b&b`	the text between these codes centered on the page; text after the second `&b` will be right-justified

Using Reports to Send E-mail

You can design a report that will create the body of an e-mail message. The report itself controls the e-mail content. An `EmailInfo` element within the `Report` element provides the parameters used to send the e-mail.

Example of an <EmailInfo> element

```
<inmg:Report Name="EmailSerialsClaim" ShowItemNumber="false" PageSize="20"
PageBreak="true" GroupBy="SuppName.SerSuppID" SortMethod="specify"
SortBy="SuppName.SerSuppID">
   <EmailInfo From="[E-mail]" ToField="SuppEmail.SerSuppID"
ToName="SuppName.SerSuppID" Subject="Overdue Issue Claim" FontName="Arial"
FontSize="10" />
…
```

The following table lists the supported attributes for the `EmailInfo` element.

Attribute	Required	Default	Explanation
`ToAddress`	No *	none	Text e-mail address of recipient.
`ToField`	No *	none	Name of field that will provide the e-mail address(es) of the recipient(s).
`ToName`	No	“??”	Name of field containing the name of the recipient. Used for error reporting.
`From`	Yes	None	E-mail address of sender. Can use a placeholder for library information, such as [E-mail].
`Subject`	Yes	None	E-mail subject line.
`FontName`	No	None	Specify the name of the font to use for the e-mail text. Separate alternatives with commas: “Verdana, Arial”.
`FontSize`	No	None	Specify the size of the font in points.
`AttachField`	No	None	Name of field providing name of file(s) to be attached to the e-mail.
`CC`	No	None	E-mail address of person to receive a copy of the e-mail.
`BCC`	No	None	E-mail address of person to receive a blind copy of the e-mail.

Notes:

- The ToField is typically a field in a linked textbase, such as SuppEmail@SerSuppID, or BorrEmail@LoanBorrID. However, in a special case such as a routing list for a serial subscription, there are no links to the Borrower textbase, and the routing list field can contain multiple names. There is a special syntax for this case: Field1:Field2, where Field1 is the name of the field in the primary textbase containing the name(s) of the recipient(s), and Field 2 is the name of the field in the Borrower textbase containing the e-mail addresses of the recipients. Example: SerRouteList:BorrEmail.
- The text of the e-mail can contain placeholders for information entered on the Set Up E-mail Information page. These are:

  [LibraryName]
  [E-mail]
  [Address]
  [Telephone]
  [WebSite]

Updating Records After Processing

After an action is performed involving the records or selected records in a report, the records that were successfully processed can be updated to reflect the action taken. An `ActionInfo` element in the `Report` element provides the parameters.

Example of an <ActionInfo> element

```
<inmg:Report Name="EmailSerialsClaim" ShowItemNumber="false" PageSize="20"
PageBreak="true" GroupBy="SuppName.SerSuppID" SortMethod="specify"
SortBy="SuppName.SerSuppID">
   <EmailInfo From="[E-mail]" ToField="SuppEmail.SerSuppID"
ToName="SuppName.SerSuppID" Subject="Overdue Issue Claim" />
   <ActionInfo Type="Update" Field="SerClaimHistory" UpdateAction="insert" Text="{d}:
sent e-mail" />
…
```

The following table lists the supported attributes for the `ActionInfo` element.

Attribute	Required	Default	Explanation
`Type`	Yes	none	Type of action to be performed. Currently only "`Update`" is supported.
`Field`	Yes	none	Name of field to receive the update text.

`Action`	No	"`substitute`"	The action to be performed on the field. Possible values are:
			`insert`
			`append`
			`substitute`
			`delete`

`Entry`	No	"`first`"	The field entry to be affected by the update. Possible values are:
			`all`
			`first`
			`last`
			`matching`

`Text`	Yes	none	The text to be placed in the field. Use {d} as a placeholder for the current date.
`OldText`	No	none	Text to be modified; use with Entry="matching".
`Trigger`	No	none	Identifies the button that causes this action to be performed. Needed only if the report contains more than one ActionInfo element.

Notes:

- If used in a report that sends e-mail, only records actually sent are updated. Note that a record can be sent if the addresses of senders and recipients are well-formed. The application has no way of knowing if any of the addresses are valid or if the e-mail ultimately reached the intended destination.
- If used in a report that contains a button to trigger the action, all selected records are updated.

MyEditScreens.CONFIG File

The MyEditScreens.CONFIG file defines all the Edit pages provided for the application. You can use this configuration file to add, modify or remove screen elements from an edit screen. You cannot, however, define additional edit screens.

MyEditScreens.CONFIG has a top level element, **EditScreens**, which is the container for the individual **EditScreen** elements. Each edit screen has a name that associates it with the page that will contain it. For example, the loans_edit.aspx page, used to edit existing Loans records, uses the "loans_edit" edit screen, whereas the loans checkout page, named loans_checkout.aspx, uses the "loans_checkout" edit screen. There is one exception to this rule: The Orders edit page, called orders_edit.aspx, will use either the "orders_item_edit" edit screen or the "orders_serial_edit" edit screen, depending on the menu item that was selected under the Orders heading.

An edit screen can consist of the following types of elements:

- EditScreen
- Toolbar
- Button
- EditSection
- SubSection
- Fields
- EditTips
- ItemsGrid

The following sections describe each of the elements listed above. Note that some style attributes have been omitted from the XML examples, for improved clarity.

The EditScreen Element

The **EditScreen** element defines the name and title of the screen. It can include **Toolbar**, **EditSection,** and **EditTips** child elements.

The following table lists the supported attributes for the **EditScreen** element, in general.

Attribute	Required	Default	Explanation
`Name`	Yes	None	Identifies the name of the screen. This is the name that the application uses to load appropriate screen components. Do not change the existing names.
`TitleForAdd`	No	None	Title that appears when the screen is being used to add a new record.
`TitleForEdit`	No	None	Title that appears when the screen is being used to edit an existing record.

The following table lists additional supported attributes for the **EditScreen** element when used to edit items.

Attribute	Required	Default	Explanation
`AllowPaging`	No	`"false"`	Set to `"true"` to support paging through the items in the items table. Use if it's possible to have many items associated with a single Catalog record.
`PageSize`	No	`10`	Number of rows to display per page.

Example of an EditScreen element:

```
<inmg:EditScreen Name="borrower_edit" TitleForAdd="Add New Borrower"
TitleForEdit="Edit Borrower Record" >
   <Toolbar Title="Data Entry Toolbar">
      ....
   </Toolbar>
   <EditSection Title="Borrower Fields">
      ...
   </EditSection>
</inmg:EditScreen>
```

The following illustration shows the screen title generated based on the above example.

Note: The heading "**Inmagic Genie:**" is configured in MyGenie.CONFIG. It is consistent throughout the Genie application. See page 25.

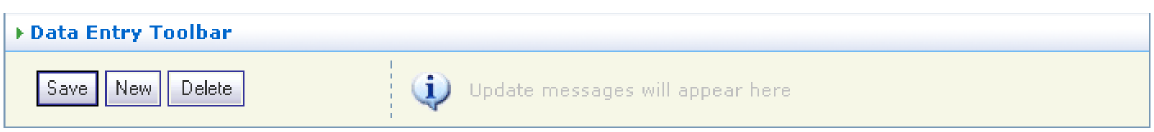
Toolbar Element

The **Toolbar** element defines the components of the toolbar section. Use the **Title** attribute to specify the title of the toolbar. Use the optional **Location** attribute to specify “top” (the default) or “bottom” (to place the toolbar below the edit boxes). Use the **Button** element to add buttons to the toolbar. The only child element that can be included in the toolbar is **Button**.

Example of a Toolbar element:

```
<Toolbar Title="Data Entry Toolbar">
   <Button Name="Add" Text="Add Record"/>
   <Button Name="Save" Text="Save"/>
   <Button Name="New" Text="New"/>
   <Button Name="Delete Text="Delete"/>
</Toolbar>
```

The following illustration shows the toolbar generated based on the above example configuration.



Note: When the screen is being used to add a new record, only the **Add** button will be displayed.

When the screen is being used to edit an existing record, the **Add** button will not appear but the other three buttons defined in the above example code do appear, as shown above.

Button

The **Button** element defines an action button to include on an edit screen. Button elements can be included in a Toolbar, or elsewhere on the page. Use the **Text** attribute to specify the text that will appear on the button, and the **Name** attribute to associate a predefined action with the button.

The following table lists the supported attributes for the **Button** element.

Attribute	Required	Default	Explanation
`ButtonWidth`	No	Auto	Specifies the width of the button. Applies only to buttons not located in the Toolbar.
`Name`	Yes	None	Specifies the name of predefined button. See the table below.
`NewRow`	No	false	If set to `"true"`, will force the button to appear in a new row. Applies only to buttons located in the Toolbar.
`Text`	No	None	Specifies the text to display on the button.
`Width`	No	Auto	Each button is wrapped in a table cell. The **Width** attribute specifies the width of the table cell. You can use this to adjust the space between the buttons. Applies only to buttons not located in the Toolbar.

The following table lists button names that can be included in the **Toolbar** of most edit pages (exclusions are listed)**.**

Button Name	All Edit Pages Except	Explanation
`Add`	loans checkin loans checkout loans edit	Add a new record to the textbase.
`Delete`	loans checkin loans checkout serials setup reserves edit	Delete the record in the textbase.
`Duplicate`	loans checkin loans checkout loans edit orders edit reserves edit	Duplicate a record. Will open the existing record in a new record window, discarding the ID field value so that a new record will be created when the record is saved. *Continued…*

Button Name	All Edit Pages Except	Explanation
`New`	loans checkin loans checkout serials edit serials setup reserves edit	Display an empty edit page to create a new record.
`NextRecord`	items edit orders edit reserves edit	Navigate to the next record in the set.
`PrevRecord`	items edit orders edit reserves edit	Navigate to the previous record in the set.
`Save`	loans checkin loans checkout reserves edit	Save updated record in textbase.

The following table lists buttons that can be used only in the specified edit pages:

Button Name	Only These Pages	Explanation
`CheckIn`	loans checkin	Check in loans.
`CheckOut`	loans checkout opac checkout	Check out loans.
`Reset`	loans checkin loans checkout opac checkout	Reset loans checkin/checkout page.
`GenerateDates`	serials setup	Generate expected arrival dates for serial issues.
`GenerateItems`	catalog edit	Open the items generation page in a new window.
`CreateItems`	items generate	Create a specified number of item records with content generated from the boxes on the page.
`Close`	items edit items generate reserves edit serials setup	Close the popup windows. (In the case of the serials setup page, this button only appears if that page was opened in a new window from the Catalog edit screen.)
`EditItem`	catalog edit serials edit	Open the item edit page in a new window.
`SaveAll`	serials setup	If multiple Serial records are found linked to the same Catalog record, this button causes all of these records to be updated with the newly generated expected arrival dates.
`Setup`	serials edit	Switch to the serials setup page.

The following table lists the buttons that can be included in the **Item** or **Serial** subsections of an orders edit page**.**

Button Name	Explanation
`AddItem`	Add a new Item or Serial.
`NewItem`	Display empty Item or Serial section to create a new Item or Serial.
`UpdateItem`	Update an existing Item or Serial.
`DeleteItem`	Delete an existing Item or Serial.

The following table lists the buttons that can be included in the **ItemReceiving** or **SerialReceiving** subsections of an orders edit page**.**

Button Name	Explanation
`MarkReceived`	Mark an Item or Serial received by filling in the date received field and setting the status to received.
`CheckCatalog`	Open check catalog window.
`CreateCatalog`	Create new catalog record.
`CreateItem`	Create new Items records.

EditSection Element

The **EditSection** element defines the edit fields and their layout. You can include a **Field** element and/or **SubSection** element as the components of the edit section. **An edit screen must have one and only one EditSection.**

The following table lists the supported attributes for the **EditSection** element.

Attribute	Required	Default	Explanation
`Title`	No	None	Title displayed in the edit section title bar. This attribute only applies to screens without subsections, or with subsections of type "Table**"**. Does not apply to the **TabStrip** section.
`SectionType`	No	Table	Supported types: **Table** or **TabStrip** Note that it is not possible to dynamically hide or show tabs. If tabs are used on an orders edit page, for example, the receiving section will always appear.

SubSection Element

Use the **SubSection** element to group fields into subsections with subtitle bars or tabs.

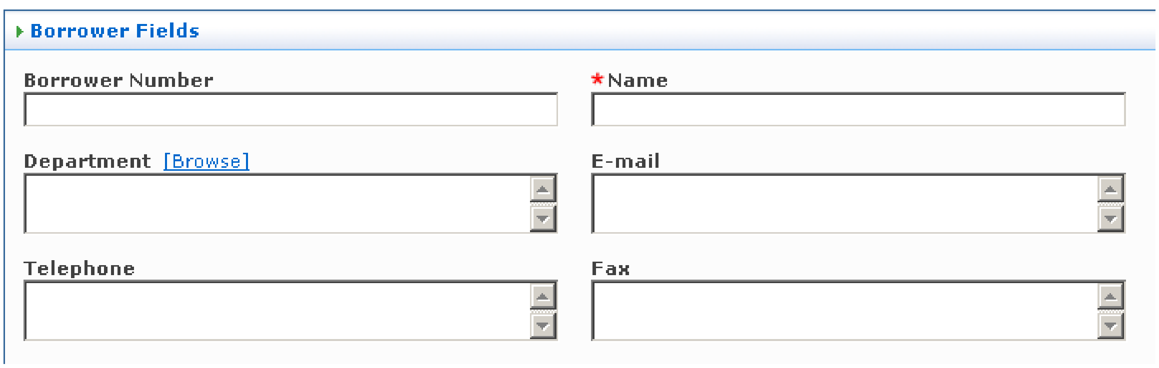
The following table lists the supported attributes for the **SubSection** element.

Attribute	Required	Default	Explanation
`Name`	Yes	None	Name of a subsection. ***Important!*** The following subsection names for the order edit page are reserved. If you change them, the application will not work properly. **Order**, **Item**, **ItemReceiving**, **Serial**, **SerialReceiving**
`Title`	No	None	Title displayed on the subsection title bar or tab. When **SectionType**="Table" and the **Title** attribute is not specified, no subsection title bar will be displayed.
`HideSection`	No	false	Only applies to **SectionType**="Table". When set to true, a Hide/Show toggle link will appear in the subsection title bar that allows you to hide/show the section.

Example of EditSection without SubSection:

```
<EditSection Title="Borrower Fields">
   <Field Style= "font-weight: bold">%%StartTable</Field>
   <Field>%%StartRow</Field>
   <Field Width="50%" Name="BorrNumber" Type="EditBox"/>
   <Field Width="50%" Name="BorrName" Type="EditBox" Required="true"/>
   <Field>%%EndRow</Field>
   <Field>%%StartRow</Field>
   <Field Name="BorrDepartment" Type="EditBox" ShowBrowse="true" NumLines="2"/>
   <Field Name="BorrEmail" Type="EditBox" NumLines="2"/>
   <Field>%%EndRow</Field>
   <Field>%%StartRow</Field>
   <Field Name="BorrTelephone" Type="EditBox" NumLines="2"/>
   <Field Name="BorrFax" Type="EditBox" NumLines="2"/>
   <Field>%%EndRow</Field>
   <Field>%%EndTable</Field>
   ...
</EditSection>
```

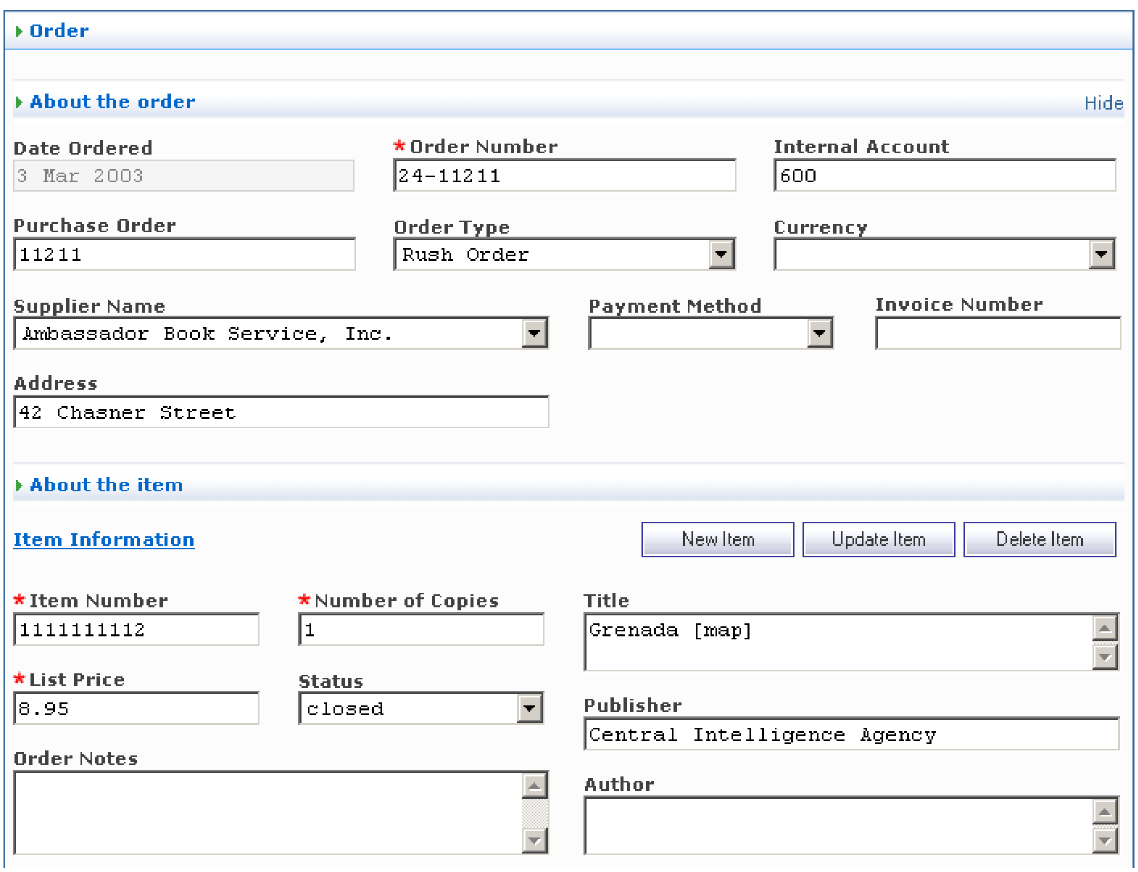
The following illustration shows the edit section generated based on the sample code above.



Example of EditSection with SubSection (SectionType="Table"):

```
<EditSection Title="Order">
   <SubSection Name="Order" Title="About the order" HideSection="true">
      <Field Align="center">%%StartTable</Field>
      <Field>%%StartRow</Field>
      <Field Name="OrdDateOrdered" Type="EditBox" Width="232" ReadOnly="true"/>
      <Field Name="OrdOrderNumber" Type="EditBox" Width="232"/>
      <Field Name="OrdInternalAccount" Type="EditBox" Width="214"/>
      <Field>%%EndRow</Field>
      <Field>%%StartRow</Field>
      <Field Name="OrdPurchaseOrder" Type="EditBox" Width="232"/>
      <Field Name="OrdOrderType" Type="DropDownList" Width="232"/>
      <Field Name="OrdCurrency" Type="DropDownList" Width="214"/>
      <Field>%%EndRow</Field>
      <Field>%%EndTable</Field>
      ...
   </SubSection>
   <SubSection Name="Item" Title="About the item">
      <Field Align="center">%%StartTable</Field>
      <Field Height="38">%%StartRow</Field>
      <Field Display="Item Information" Width="300">%%Label</Field>
      <Button Name="AddItem" Text="Add to Item List"></Button>
      <Button Name="NewItem" Text="New Item"></Button>
      <Button Name="UpdateItem" Text="Update Item"></Button>
      <Button Name="DeleteItem" Text="Delete Item></Button>
      <Field>%%EndRow</Field>
      <Field>%%EndTable</Field>
      ...
   </SubSection>
</EditSection>
```

The following illustration shows the edit section with subsection (SectionType="Table") generated based on the sample code above.

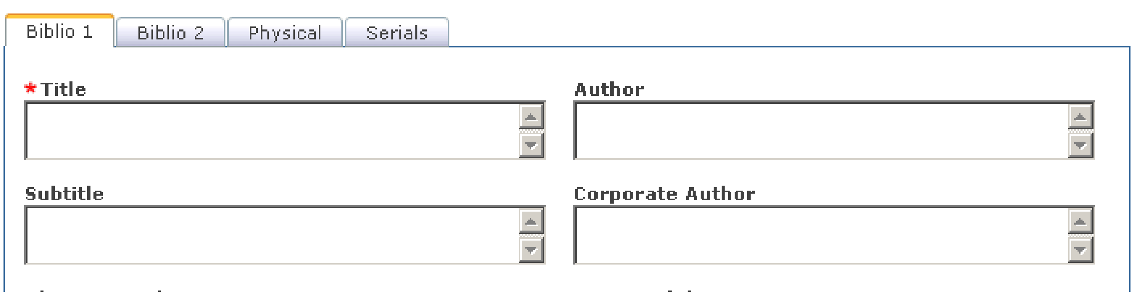


Example of EditSection with SubSection (SectionType="TabStrip"):

```
<EditSection SectionType="TabStrip">
   <SubSection Name="Biblio1" Title="Biblio 1">
      <Field>%%StartTable</Field>
      <Field>%%StartRow</Field>
      <Field Width="50%" Name="CatTitle" Type="EditBox" Required="true"
      NumLines="2"/>
      <Field Width="50%" Name="CatAuthor" Type="EditBox" NumLines="2"/>
      <Field>%%EndRow</Field>
      <Field>%%StartRow</Field>
      <Field Name="CatSubtitle" Type="EditBox" NumLines="2"/>
       <Field Name="CatCorporateAuthor" Type="EditBox" NumLines="2"/>
      <Field>%%EndRow</Field>
      ...
   </SubSection>
   <SubSection Name="Biblio2" Title="Biblio 2">
      <Field>%%StartTable</Field>
      <Field>%%StartRow</Field>
      <Field Name="CatSubjects" Type="EditBox" ShowBrowse="true" NumLines="6"/>
      <Field Name="CatAbstract" Type="EditBox" NumLines="6"/>
      <Field>%%EndRow</Field>
      <Field>%%StartRow</Field>
      <Field Name="CatISBN" Type="EditBox" NumLines="1" TextMode="MultiLine"/>
      <Field Name="CatISSN" Type="EditBox" NumLines="1" TextMode="MultiLine"/>
      <Field>%%EndRow</Field>
      ...
   </SubSection>
   <SubSection Name="Physical" Title="Physical">
      <Field>%%StartTable</Field>
      <Field>%%StartRow</Field>
      <Field Name="CatRecordType" Type="EditBox" ShowBrowse="true" NumLines="1"/>
      <Field Name="CatStatus" Type="EditBox" ShowBrowse="true" NumLines="1"/>
      <Field>%%EndRow</Field>
      <Field>%%StartRow</Field>
      <Field Name="CatCallNumber" Type="EditBox" NumLines="1" TextMode="MultiLine"/>
      <Field Name="CatDateCataloged" Type="EditBox" NumLines="1"
      ShowCalendar="true"/>
      <Field>%%EndRow</Field>
      ...
   </SubSection>
   ...

</EditSection>
```

The following illustration shows the edit section with subsections (**SubSection**="TabStrip") generated based on the example code above.



Field Element

The **Field** element can be a data field which has a **Name** attribute corresponding to the **AppName** defined in MyGenieFieldMaps.CONFIG, a formatting field with special tags such as **%%StartTable** or **%%StartRow**, or a special purpose tag for specific purposes such as %%ViewItems **(**see the example showing the loans checkout screen on page 97).

The following tables describe the attributes supported by field elements.

Attribute	Req'd	Default	Explanation
`AutoIncrement`	No	false	Used in the items_generate screen to specify that the field value should increase by 1 for each new item record created.
`BrowseField`	No	The field specified in the **Name** attribute	Specify the name of the field to browse.
`BrowseType`	No	VALIDATION_LIST	Specify the browse window content type. Supported values: **TERM_INDEX**, **VALIDATION_LIST**, **FIELD_VALUE** (displays actual field value from textbase)
`BrowseTN`	No	Current textbase (e.g for order page, it is Orders, for Catalog page, it is Catalog)	Specify the textbase which you want to browse.
`CheckIn`	No	false	Specify whether to show the **CheckIn** link or not. Only applies to serials edit page
`ClearOnDuplicate`	No	False	Specifies that if the record is duplicated, this field value should not be carried forward to the duplicated copy.
`ClearRouteDate`	No	false	Specify whether to show the "Clear" hyperlink. Only applies to the serials edit page
`CssClass`	No	No	CSS class applies to field contents (e.g. **TextBox**, **DropDownList**). Read from the Genie.CSS file (or MyGenie.CSS) in the root of the *Genie* installation folder.
`Display`	No	If used for TextBox or DropDownList field, it maps to the BoxLabel attribute in the MyGenieFieldMaps .CONFIG file. Otherwise, none.	When used for BoxLabel display, it overrides what is specified in the MyGenieFieldMaps.CONFIG file. It is also used for special links display and %%Label tag display, and **Type="Label"** field display.
`DisplayCss`	No	None	Each display field is wrapped in a container table cell. The DisplayCss is applied to the container table, table row and table cell. It in turn affects the display of the BoxLabel.
`ExtractTextDisplay`	No	None	Used with the %%FileUpload special tag to specify the label for the "Extract Text" CheckBox.

`ImageURL`	No	None	Specify the location of the image to be displayed. Only applies to the **EditTips** element and the following special tags: **%%OverdueNotice** **%%PrintRoutingNotice** **%%EmailRoutingNotice**
`Name`	Yes, unless used with special tags	None	Should be the same as the AppName defined in the MyGenieFieldMaps.CONFIG file. ***Note:*** If the field you specify is a secondary field, it must be read-only.
`NumLines`	No	1	Specify the number of lines of a TextBox.
`Persists`	No	false	Set to "true" if you want the content of this field to be remembered, and used to pre-populate this box the next time a new record is created. (The content is remembered in a cookie on the machine running the browser session.)
`ReadOnly`	No	false	Applies to the fields with **Type**="TextBox" to specify that the TextBox should be read-only.
`Required`	No	false	Specify if a field is a required field or not. ***Note:*** LoanItemBC, LoanBorrID and LoanTerm are always required for the loans check out page. LoanTerm is always required for the loans edit page. The Required attribute cannot be turned off for these fields.
`SkipMissingIssues`	No	false	Specify whether to show the "Skip missing issue" hyperlink or not. Only applies to the serials edit page
`ShowBrowse`	No	false	Specify whether to show the browse link for the field. ***Note:*** This attribute cannot be applied to boxes having **Type**="DropDownList".
`ShowCalendar`	No	false	Specify whether to show the Calendar button.
`Style`	No	None	HTML style applied to display labels and container table cells.
`TabIndex`	No	0	Specify tab order, 1 to N. Use -1 to skip the box when tabbing. 0 means tab in paint order.

`TextMode`	No	MultiLine if NumLines > 1, otherwise SingleLine	Applies to fields with **Type**="TextBox" to specify the textmode of the TextBox. Use TextMode="MultiLine" to display a scrollbar on a textbox set to 1 line high.
`ToolTip`	No	None	Specify optional tool tip that appears if the user places the cursor over the text box.
`Type`	No	TextBox	Type of the display UI for the field. Supported values: **TextBox**, **DropDownList**, **Label**. ***Note:*** You can only use the DropDownList Type for fields already implemented in Genie with that Type. Such fields have support in Genie for populating the items on the list.
`UploadBoxSize`	No	35	Applies to the **%%FileUpload** special tag to specify the size of the file upload box, when used with **UploadType**="BookWhere".
`UploadType`	No	Image	Type of upload file, for use with the **%%FileUpload** special tag. Supported values: **BookWhere, Image, Document, ToCImage, ToCDocument**. Controls which folder the file will be placed in on the server.
The following attributes are related to tables			
`Align`	No	None	Applies to the **%%StartTable** and **%%StartSubTable** special tags and individual **Field** to specify the cell text alignment. Values can be **Left**, **Right**, and **Center**.
`CellPadding`	No	0	Applies to the **%%StartTable** and **%%StartSubTable** special tags to specify the CellPadding for the table.
`CellSpacing`	No	0	Applies to the **%%StartTable** and **%%StartSubTable** special tags to specify the CellSpacing for the table.
`ColSpan`	No	1	Specify the number of columns a field will occupy. If you change this value, you may need to change the CssClass as well, if it specifies a conflicting width. See page 111.
`Height`	No	Auto	Applies to the **%%StartRow** special tag to specify the height of the table row.
`VAlign`	No	Middle	Applies to the **%%StartRow** special tag to specify the row vertical alignment. Values can be **Top**, **Bottom**, and **Middle**.
`Width`	No	50%	Applies to controls on the edit page, such as **%%StartTable**, **%%StartSubTable**, and **Field** elements.

Special Tags

The following table lists predefined special tags that you can use in the edit screen configuration. You enter predefined tags as a field element's inner text in the MyEditScreens.CONFIG file. Note that the text identifying a text box or linkbutton described in these tables can generally be changed, using the Display attribute on the Field element.

Special Tags for Use on All Edit Screens

Tag	Explanation
%%StartTable	Used in tabular layout. Marks the beginning of an HTML table. Any fields inserted after this and before the **%%EndTable** tag will be displayed inside this table.
%%EndTable	Marks the end of a table.
%%StartSubTable	Used in tabular layout. Marks the beginning of a HTML table nested inside the table defined by **%%StartTable**.
%%EndSubTable	Marks the end of a subtable.
%%StartRow	Marks the beginning of a table row. Any fields inserted after this and before the **%%EndRow** tag will be displayed inside the row. It must be within the **%%StartTable** and **%%EndTable** block or within the **%%StartSubTable** and **%%EndSubTable** block.
%%EndRow	Marks the end of a row.
%%Label	Displays any text and/or HTML tags that are not related to a textbase field

Special Tags for Catalog Edit Screens

Tag	Explanation
%%CheckForSerials	Display "Check for Serial Records" LinkButton.
%%CreateSerial	Display "Create Serial Record" LinkButton.
%%EditSerial	Display "Edit Serial Record" LinkButton. Only appears after the Create Serial Record LinkButton has been clicked.
%%SerialRecords	Works with **%%CheckForSerials** and **%%CreateSerial** special tags, displays "Corresponding Serial Records" read-only textbox.

Special Tags for Serials Edit Screens

Tag	Explanation
%%CreateItem	Display a link to create an item record for the issue just checked in.
%%EmailRoutingNotice	Display a "Send e-mail routing notices" hyperlink.
%%PrintRoutingNotice	Display "Print routing notices" Hyperlink.
%%ToCImageUpload	Display Table of Contents image upload control.
%%ToCImageUploadHelp	Display help text area for Table of Contents image upload control.

Special Tags for Serials Setup and Generate Items Pages

Tag	Explanation
%%IssuesPerVolume	Display an "Issues per Volume" textbox.
%%SerialDuration	Display a "Duration (months)" textbox. Applies to the Serials Setup page only.
%%StartIssue	Display a "Starting Issue" textbox.
%%StartVolume	Display a "Starting Volume" textbox.
%%NumberOfItems	Display a "Number of Items" textbox. Applies to a Generate Items page only.

Special Tags for the LoansCheckout Page

Tag	Explanation
%%BorrowerInfo	Provides an area on the page where Borrower information appears.
%%ItemsFeedback	Provides an area on the page where Items information appears.
%%ViewBorrower	Display a "Borrower" linkbutton. OPTIONAL; auto-displays in v3.2.
%%ViewItems	Display an "Items" linkbutton. OPTIONAL; auto-displays in v3.2.
%%ViewReserves	Display a "Reserves" linkbutton. DEPRECATED; auto-displays in v3.2.

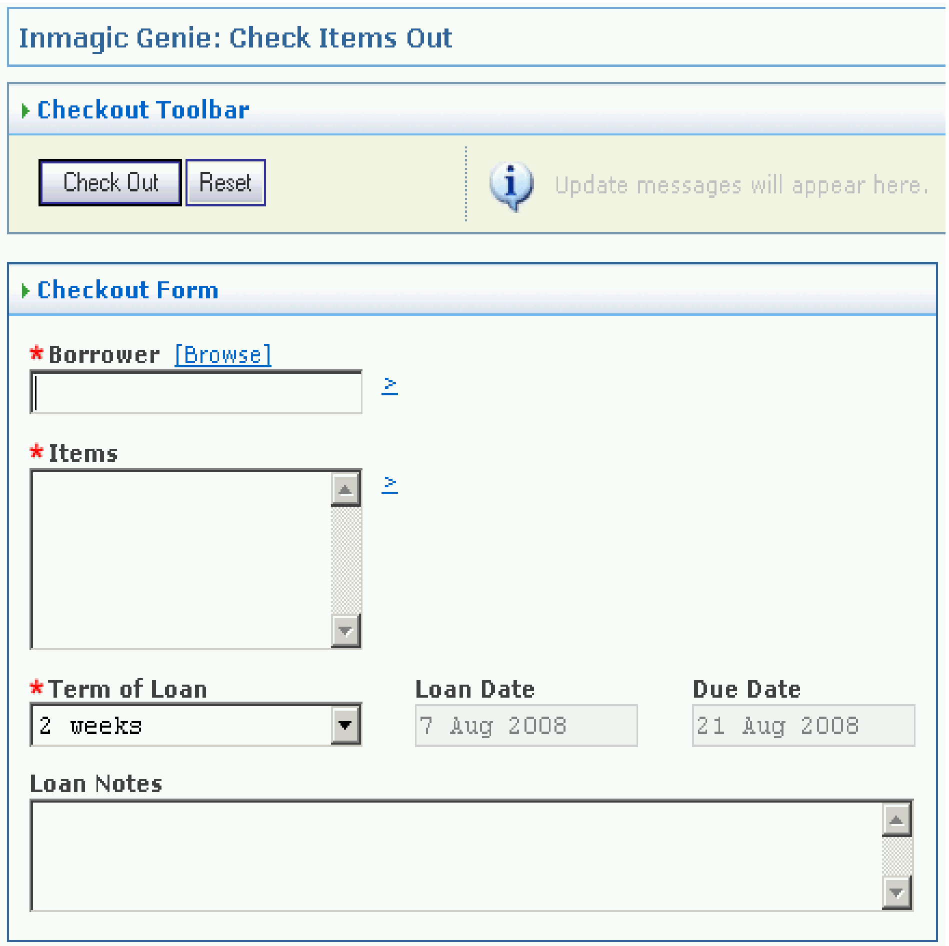
Special Tags for the Loan Edit Screen

Tag	Explanation
%%LoanItemInfo	Display a read-only textbox that will display information about the item on loan.
%%OverdueNotice	Display a "Send overdue e-mail notice" linkbutton.

Example from the loans checkout screen:

```
   <inmg:EditScreen Name="loans_checkout" TitleForAdd="Check Items Out"
TitleCss="tblheader">
      <Toolbar Title="Checkout Toolbar">
         <Button Name="CheckOut" Text="Check Out"/>
         <Button Name="Reset" Text="Reset"/>
      </Toolbar>
      <EditSection Title="Checkout Form" Width="500">
         <Field Style= "font-weight: bold; padding-bottom: 6; padding-top: 6"
CellSpacing="3" CellPadding="3" Align="center" Width="100%">%%StartTable</Field>
         <Field VAlign="top">%%StartRow</Field>
         <Field Name="LoanBorrID" Display="Borrower" Required="true" ShowBrowse="true"
CssClass="loans_checkout_textinput" BrowseTN="Borrower" BrowseField="BorrName"
BrowseType="TERM_INDEX"/>
         <Field Display="&lt;br /&gt;>" Style="font-weight:normal" Width="20"
Align="left" CellPadding="6">%%ViewBorrower</Field>
         <Field ColSpan="2">%%BorrowerInfo</Field>
         <Field>%%EndRow</Field>
         <Field VAlign="top">%%StartRow</Field>
         <Field Name="LoanItemBC" Display="Items" Required="true" NumLines="5"
CssClass="loans_checkout_textinput" BrowseTN="Items" BrowseField="ItemBarCode"
BrowseType="TERM_INDEX"/>
         <Field Display="&lt;br /&gt; > " Style="font-weight:normal" Width="20"
Align="left" CellPadding="5">%%ViewItems</Field>
         <Field ColSpan="2">%%ItemsFeedback</Field>
         <Field>%%EndRow</Field>
         <Field VAlign="top">%%StartRow</Field>
         <Field Name="LoanTerm" Required="true" Type="DropDownList"
CssClass="loans_checkout_textinput"/>
         <Field Display="" Style="font-weight:normal" Width="20"
Align="left">%%Label</Field>
         <Field Name="LoanDateLoaned" CssClass="loans_checkout_readonly_box"
ReadOnly="true"/>
         <Field Name="LoanDateDue" Display="Due Date"
CssClass="loans_checkout_readonly_box" ReadOnly="true" />
         <Field>%%EndRow</Field>
         <Field>%%StartRow</Field>
         <Field Name="LoanNotes" ColSpan="4" CssClass="item_edit_wide_textinput"
NumLines="3"/>
         <Field>%%EndRow</Field>
         <Field>%%EndTable</Field>
      </EditSection>
      <EditTips ShowIcon="true"  Title="How To Check Out"
ImageURL="Images/xp_bulb.gif"
             TipContent="&lt;ol style=&quot;margin-left:25px;&quot;&gt;&lt;li&gt;Enter
the borrower name or number.&lt;/li&gt;&lt;li&gt;
       Type or scan item bar codes into the Items box, each on a separate line.
&lt;/li&gt;&lt;li&gt;
       Select a loan term from the Term of Loan drop-down list.&lt;/li&gt;&lt;li&gt;
       Type any loan notes necessary. Separate field entries with a carriage return
and an asterisk, for example:&lt;br&gt;  &amp;nbsp;*entry 1 &lt;br&gt;
&amp;nbsp;*entry 2&lt;br&gt;&lt;/li&gt;&lt;li&gt;
       Click the &lt;b&gt;Check Out&lt;/b&gt; button.&lt;/li&gt;&lt;/ol&gt;"/>
   </inmg:EditScreen>
```

The following illustration shows the result:



ItemsGrid Element

The **ItemsGrid** element, available for the Items edit screen, specifies the fields that appear in the table and the order in which the items are displayed.

One attribute is supported for the **ItemsGrid** element.

Attribute	Required	Default	Explanation
`SortBy`	No	“`ItemID`”	Field or fields by which to sort the rows. Separate multiple fields with | if needed. Precede a field name with a hyphen for a reverse sort.

The ItemsGrid element can contain multiple Field elements, each representing a column in the table. These attributes are supported for a Field element.

Attribute	Required	Default	Explanation
`Name`	Yes	None	Name of field to appear in the column.
`Label`	No	Default field label	Column label.
`Width`	No	None	Column width.

Uploading Files

File upload controls can be added to edit screens for specific textbases. A maximum of one document upload control can be added to a Catalog or Serials edit screen. No document upload controls can be added for other textbases. A maximum of one image upload control can be added to any textbase except Items, Reserves, or Orders. Only one document and/or one image can be referenced in a record.

To add a file upload control, use the following special tag: `%%FileUpload`

Then use these attributes:

Attribute	Required	Default	Explanation
`Name`	Yes	None	Specifies the field that will contain the name of the uploaded file.
`UploadType`	Yes	Image	Sets the upload type. Can be any of the following values: • `BookWhere` • `Document` • `Image` • `ToCDocument` • `ToCImage` See the notes that follow, describing what these values mean.
`UploadBoxSize`	No	35	Specifies the size (width) in characters of the box that will contain the file name. Applies to the BookWhere UploadType only.
`Display`	No	None	Text for the label that appears above the controls.
`DisplayCss`	No		Style for the label text.
`NewLinkText`	No	"Upload…"	Text for the link to upload a new file.
`ReplaceLinkText`	No	"Replace…"	Text for the link to upload a file that will replace the one already there.
`RemoveLinkText`	No	"Remove"	Text for the link to remove the file from the field.
`NewHelpText`	No	None	Help that appears when the field has no content.
`ReplaceHelpText`	No	None	Help that appears when the field already has content.
`HelpDisplayCss`	No	None	Style for the help text.

The UploadType governs the folder into which the file will be uploaded. Note that to extract the text from a document and make it searchable, the *Importer* service must be running, and be set up to watch the specified import folders and have the appropriate import parameters set up for each.

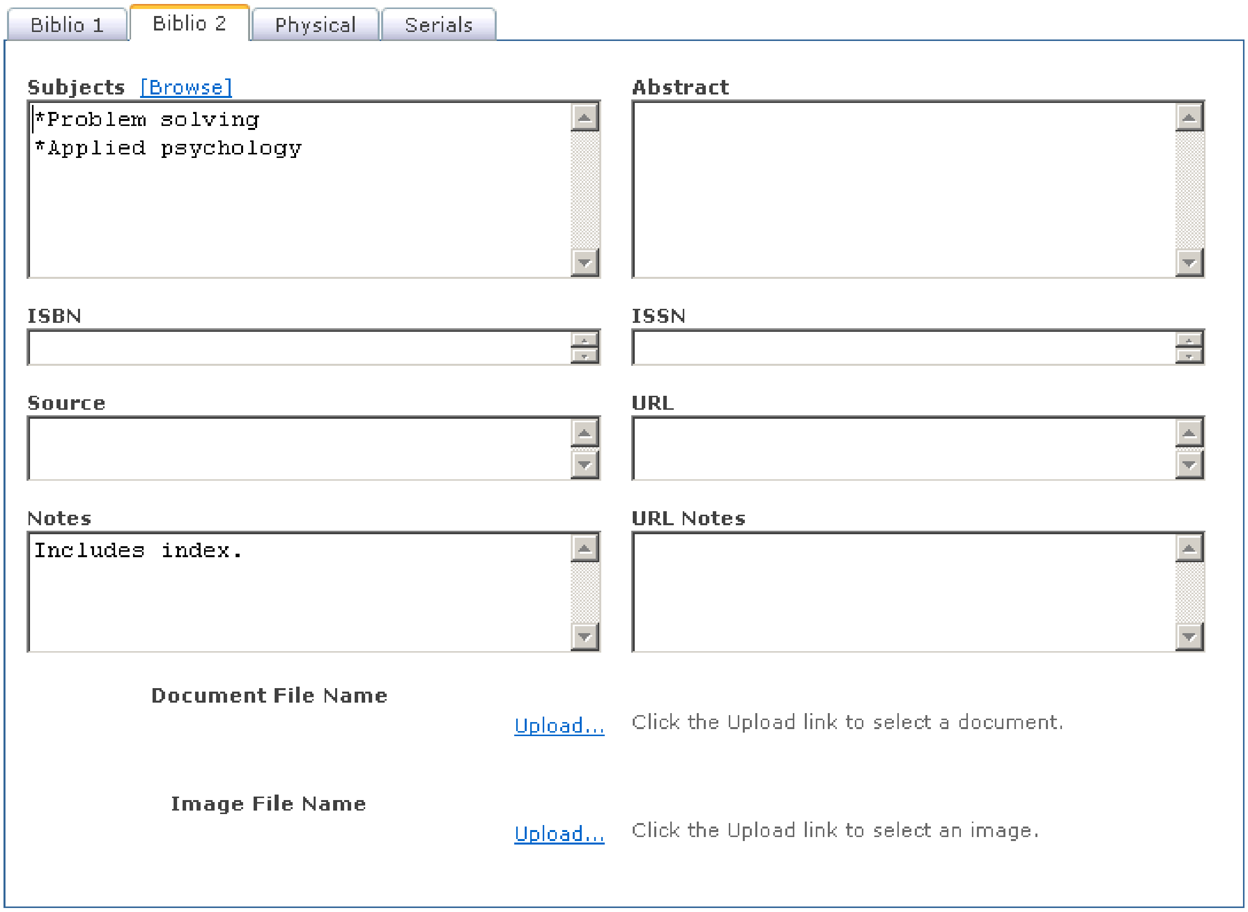
Type	Explanation
`BookWhere`	This file will be saved to the folder specified in the **BookWhereFolder** key in MyGenie.CONFIG, and either moved or deleted, depending on your *Importer* setting, when the import has completed processing. Note that this `UploadType` is different from the others, as it is not adding a file name to a record in a textbase. Instead, it is adding one or more new bibliographic records to the Catalog textbase. Do not add this type of control to an edit screen. It is used in a separate page named bookwhere_upload.aspx. The following attributes are not used with this `UploadType`: `ReplaceLinkText`, `RemoveLinkText`, `ReplaceHelpText`.
`Document`	The final destination is the folder specified in the **DocumentFolder** key in MyGenie.CONFIG. If Extract Text is checked on the file upload page, this file will be saved to the folder specified in the **ImportFolder** key in MyGenie.CONFIG, and moved by the *Importer* when it has finished processing the file.
`Image`	Will be saved to the folder specified in the **ImageFolde**r key in MyGenie.CONFIG.
`ToCDocument`	The final destination is the folder specified in the **DocumentFolder** key in MyGenie.CONFIG. If Extract Text is checked on the file upload page, this file will be saved to the folder specified in the **ToCImportFolder** key in MyGenie.CONFIG, and moved by the *Importer* when processing is complete.
`ToCImage`	Will be saved to the folder specified in the **ToCImageFolder** key in MyGenie.CONFIG.

For a Catalog record, you can add controls to upload a document file and/or an image file. For a Serial record, you can add a single file upload control, for a document file or an image file, but not both.

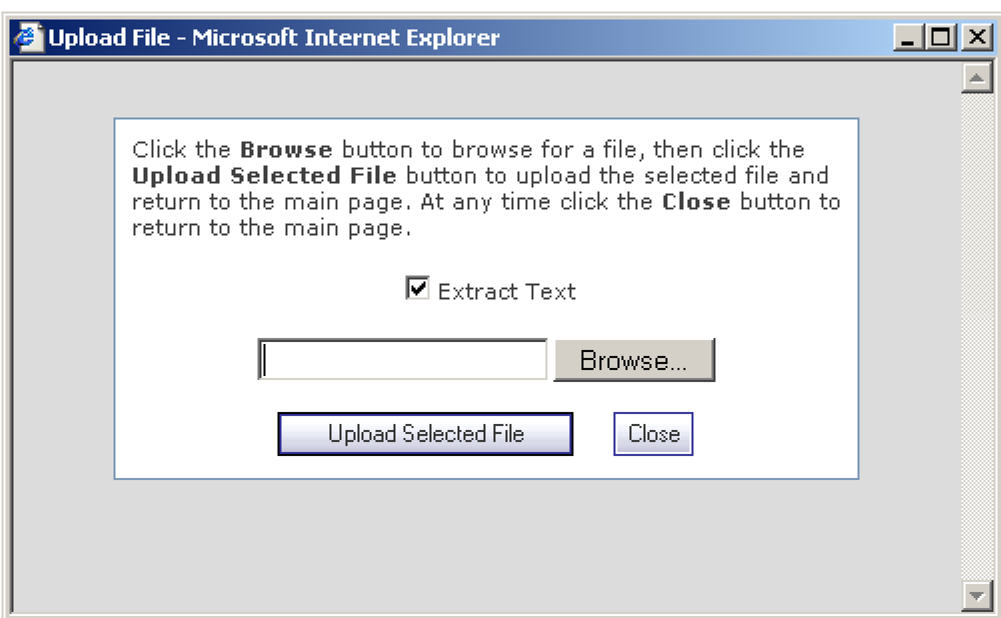
For a document file, you can link to the document and/or use, if you have it installed, the Inmagic® *Importer* to import the full text of the document. For an image file, you can link to the image file. By default, this functionality is available out-of-the-box for document files on the Catalog edit screen. See information about the upload folders and *Importer* starting on page 7.

Note that document upload links are not enabled for new records if text extraction is supported. To upload a document for a new record, save the record first. This ensures that the record is present in the textbase when the *Importer* attempts to add the extracted text. To disable the ability to extract full text from uploaded documents, comment out the **ImportFolder** and/or **ToCImportFolder** keys in MyGenie.CONFIG.

The following illustration shows the file upload functionality activated for both document and image files on the Catalog edit screen.



When you click the Upload link for a document, this popup window appears:



Note: If imported, the full text is not displayed in the *Genie* application by default, as its content could be quite large. Extracting full text is only done to support searching by word, phrase, or proximity expression.

EditTips Element

For most edit screens, the edit tips are located under the navigation menu, and default content is supplied.

For the loans checkin, loans checkout, reserves edit and generate items pages, the edit tips are located in the edit area. They only appear if specified with the EditScreen.

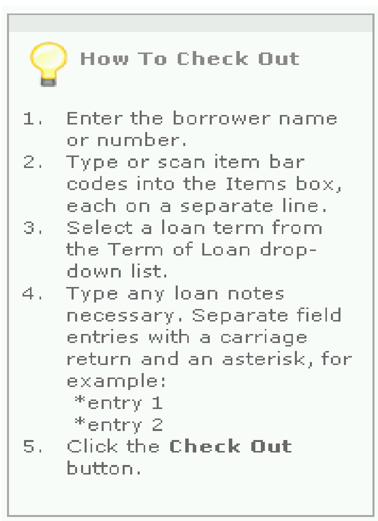
The following table lists the supported attributes for the **EditTips** Element.

Attribute	Req'd	Default	Explanation
`ImageURL`	No	`"Images/ xp_bulb.gif"`	Provides the location for the image to be used as an icon on the Edit Tips box. Use in conjuction with the `ShowIcon` attribute. The default, xp_bulb.GIF, is located in the Images subfolder of the main *Genie* installation folder. You can enter any path that the application can access.
`Location`	No	`"right"`	Set to "right" to display the tips to the right of the edit screen boxes. Set to "left" if they should replace the tips beneath the navigation menu (on any edit screen where the tips already appear there).
`ShowIcon`	No	`"true"`	Sets whether or not to show an icon before the title of the Edit Tips box. When set to "true", you must provide the path to an image file in the `ImageURL` attribute. Set to "false" if you do not want to show an icon.
`Title`	No	""	Sets the title for the Edit Tips box. If the Title attribute is missing or empty, no title will be displayed.
`TipContent`	No	None	Enter your tips text. Be sure to encode <, >, and & characters.
`Width`	No	`"192"`	Specify the width of the box in pixels. Can also be specified as a percentage, for example: `Width="30%"`.

Example of the EditTips for the loans checkout screen:

```
<EditTips ShowIcon="true" Title="How To Check Out" ImageURL="Images/xp_bulb.gif"
             TipContent="&lt;ol style=&quot;margin-left:25px;&quot;&gt;&lt;li&gt;Enter
the borrower name or number.&lt;/li&gt;&lt;li&gt;
      Type or scan item bar codes into the Items box, each on a separate line.
&lt;/li&gt;&lt;li&gt;
      Select a loan term from the Term of Loan drop-down list.&lt;/li&gt;&lt;li&gt;
      Type any loan notes necessary. Separate field entries with a carriage return
and an asterisk, for example:&lt;br&gt;  &amp;nbsp;*entry 1 &lt;br&gt;
&amp;nbsp;*entry 2&lt;br&gt;&lt;/li&gt;&lt;li&gt;
      Click the &lt;b&gt;Check Out&lt;/b&gt; button.&lt;/li&gt;&lt;/ol&gt;"/>
```

The following illustration shows the edit tips generated based on the above example configuration.



MyStatistics.CONFIG File

The MyStatistics.CONFIG file specifies what will appear on the Statistics page. This page performs a series of canned searches behind the scenes, shows the number of items found for each, and includes links to see the results and perhaps take an action.

The top-level element, **`Statistics`**, contains one child element ("**`Statistic`**") per query.

A `Statistic` element must include a **`Name`** attribute that identifies that statistic. It also contains child elements, described below.

Label Element (required)

The **Label** element specifies the display string for the query after the hit count. Example: Overdue Loans (1-7 days).

Textbase Element (required)

The **Textbase** element specifies the textbase being searched.

Query Element (required)

The **Query** element specifies the command query used to retrieve the set. It should refer to textbase fields by their AppNames (original names).

The following table lists the supported attributes for the **Query** element.

Attribute	Required	Default	Explanation
`SortBy`	No	None	Lists the fields by which the records will be sorted when sent via e-mail. For example **`SortBy="BorrName.LoanBorrID"`** sorts the set of overdue items by borrower name, ensuring that only one e-mail will be sent to each borrower, listing all of their overdue items.

View Element

The **View** element specifies the name of the report page to open when the View link is clicked, for example loansoverdue_report.aspx . This element is optional, but if it is not present, the user won't be able to see the records retrieved by the query. Note that if the query doesn't retrieve any records, the View link won't appear.

The following table lists the supported attributes for the **View** element.

Attribute	Required	Default	Explanation
`Report`	No	None	Specifies the name of a report defined in MyReports.CONFIG that you want use to display the records when the View link is clicked. This example specifies that the SerialsClaim report will be used: `<View Report="SerialsClaim"> serials_report.aspx</View>` ***Note:*** This attribute is ignored if used with the loans overdue page, which always uses the "LoansOverdue" report.

Print Element

Add a **Print** element to display a Print link for the Statistic. This element specifies the print-friendly report page to open when the Print link is clicked. Supported pages are batch_print.aspx and BookLabel.aspx. If no records are retrieved by the query, no Print link will appear.

The following table lists the supported attributes for the **Print** element.

Attribute	Required	Default	Explanation
`Report`	Only if the page is batch_print.aspx	None	Specifies the report name to use when the Print link is clicked. For example: `<Print Report="Route1on8.5x11"> batch_print.aspx</Print>`
`Label`	Only if the page is BookLabel.aspx	None	Specifies the label name to use when the **Print** link is clicked. For example: `<Print Label="LabelRoute3on8.5x11"> BookLabel.aspx</Print>`

Email Element

Add an **Email** element to display an E-mail link for the Statistic. This element specifies the name of the html template file used to format the e-mail content. The html template file works with the fields defined in the **FieldList /HeaderList** attributes of the **Email** element.

If the **Email** element is not present or the template file name is not present, no E-mail link will appear. Also, if no records are retrieved by the query, no E-mail link will appear.

The following table lists the supported attributes for the **Email** element.

Attribute	Required	Default	Explanation
`HeaderList`	No	None	List the fields to be included in the email header. Separate multiple fields with a | character. These are the fields that should appear in the header section of the HTML template. If you add a field to the template, be sure to add it to the HeaderList as well. ***Note:*** Currently, only used for `Email Type="SerialsOverdue"`.
`FieldList`	No	None	List the fields to be included in the email body. These are the fields that should appear in the body section of the HTML template.
`LabelList`	No	None	List the column headings for the LoansOverdue records table. The LabelList attribute is optional. If omitted, no column headings appear. If you do set LabelList, make sure each field listed in the FieldList has a corresponding label.
`Type`	Yes	None	Specifies the type of e-mail. Supported values: • `LoansOverdue` • `SerialsOverdue` • `RouteSerials`
`TreatToCImageAs`	No	`"ImageLink"`	Specifies whether to send ToCImage as an image link or an attachment. Supported values: • `ImageLink` • `Attachment` ***Tip:*** By default, the ImageLink display text is "View the Table of Contents". You can change the display text by appending it within parentheses after the type: `TreatToCImageAs="ImageLink(View ToC)"`
`Subject`	No	None	Specifies the subject line for the e-mail.

Note that the `Type` attribute is required. The implemented types are used by the software to determine how the records should be arranged in the e-mail messages, what field in what textbase contains the recipient's e-mail address, etc.

E-mail HTML Template File

Each Statistic can have its own e-mail template file. The template may include a placeholder for information in the retrieved records.

Template files can also include placeholders for library information. The library information is entered on the Set Up E-mail Information page and saved in the MyGenie.CONFIG file.

Supported library information placeholders are listed below.

Placeholder	Required	Default	Explanation
[LibraryName]	No	None	Insert the library name here.
[Address]	No	None	Insert the library address here.
[Telephone]	No	None	Insert the library telephone here.
[E-mail]	No	None	Insert the library e-mail address here.
[WebSite]	No	None	Insert the library Web site address here.

Requirement for LoansOverdue e-mail

The LoansOverdue e-mail template must include a placeholder [RecordData] where record data will appear. See the StatsOverdueEmail_0.htm file in the *Genie* installation folder for an example.

Requirement for SerialsOverdue e-mail

The SerialsOverdue e-mail template can include header and body sections. The header content only appears once for a single supplier. The body content repeats if more than one record is retrieved for a single supplier.

Use the HeaderList attribute to list the fields to include in the header. Each field listed in the HeaderList must have a corresponding placeholder in the header part of the template. The placeholder is the field's AppName (original name) within square brackets. For example: [SuppName.SerSuppID].

Use the FieldList attribute to list the fields to include in the body. Each field listed in the FieldList must have a corresponding placeholder in the body section of the template. The placeholder is the field's AppName (original name) within square brackets. The template must use [BodyStart] and [BodyEnd] tags to mark the beginning and end of the body section. Below is an example from the StatsSerialsClaimEmail_0.htm file in the *Genie* installation folder.

```
[BodyStart]
<tr>
<td colSpan="2">
<p>We have not received the issue(s) of

</p>
<p style="FONT-WEIGHT: bold;">[SerTitle]
</p>
<p>that we expected to receive by <strong>[SerDatesMissing]</strong>. Please let us
know when we can expect receipt. Thank you.
</p>
<p style="FONT-SIZE: 6pt; MARGIN-BOTTOM:
5px">.................................................</p>
</td>
</tr>
[BodyEnd]
```

Requirement for RouteSerials e-mail

The RouteSerials e-mail must contain a body section surrounded by the **[BodyStart]** and **[BodyEnd]** tags**.** The body section repeats if more than one record is retrieved for a single borrower.

Use the **FieldList** attributes to list the fields to include in the e-mail. Each field listed in the **FieldList** must have a corresponding placeholder in the body section of the template, consisting of the field's AppName (original name) within square brackets. See the StatsRouteEmail_0.htm file in the *Genie* installation directory for an example.

MyUserData.CONFIG File

The MyUserData.CONFIG file specifies what will appear on the My Genie page accessible to end-users. My Genie is intended to show the borrower his or her relevant information, such as contact information, outstanding loans, serials routed to them, etc. The page includes an E-mail link so that the borrower can send e-mail to the library requesting a change in contact information, renewal of specific loans, etc.

The top-level element, **`UserData`**, contains one child element ("**`UserDataQuery`**") per query.

A `UserDataQuery` element must include a **`Name`** attribute that identifies that query. It also contains child elements, described below.

Heading Element (required)

The **Heading** element specifies the heading that will appear above a group of fields or records retrieved by the query.

Textbase Element (required)

The **Textbase** element specifies the textbase being searched.

Report Element (required)

The **Report** element specifies the report used to format the records retrieved by the query. In general, the queries are performed in different textbases, so an appropriate report must be used.

Query Element (required)

The **Query** element specifies the command query used to retrieve the set. It should refer to textbase fields by their AppNames (original names). Use "[BorrID]" as a placeholder for the borrower's ID, and "[BorrName]" as a placeholder for the borrower name.

MyXxx.CSS Files

The Genie pages that include references to various Genie CSS files now also include references to My CSS files. Unlike the CONFIG files, no MyXxx.CSS files are created by the installer. Instead, the user can create them if needed, and only put in the styles that should be different from the ones shipped with Genie.

These are the stylesheet files that ship with Genie:

- **CalendarStyle.CSS** – used in the calendar popup control that can be used in edit and query screens
- **Genie.CSS** – used in most Genie screens
- **Print.CSS** – used in print-friendly reports and labels
- **TabStyle.CSS** – used in any edit screen containing tabs

For example, these are the first three styles defined in Print.CSS:

```
.batch_print_body  /* used for batch report print */
{
  MARGIN-TOP: 0in;
  FONT: 12pt Times;
  MARGIN-LEFT: 0in;
  BACKGROUND:White;
}

.book_label_body /* used for book label print page */
{
  MARGIN-TOP: 0in;
  BACKGROUND: white;
  FONT: 10pt Times;
  MARGIN-LEFT: 0in;
}

.display_head
{
  font-weight: bold;
  color: darkblue;
}
```

To change the font used in print-friendly reports from Times Roman to Arial, create a MyPrint.CSS file containing only one style:

```
.batch_print_body  /* used for batch report print */
{
  MARGIN-TOP: 0in;
  FONT: 12pt Arial;
  MARGIN-LEFT: 0in;
  BACKGROUND:White;
}
```

Application Configuration File

Important! Be careful when you edit the Web.CONFIG file. We recommend that you edit only the specific areas that we have documented for you to edit.

To set an explicit culture other than the United States for use when parsing numbers, dates, and currency in the *Genie* application, you must change the `<globalization>` section of the Web.CONFIG file. This file is located in the main *Genie* installation folder. Note that this configuration file must stay as it is named.

For example, when you set the culture to `en-GB`, your numeric dates will appear as 31/01/06 for 31 January 2006 (instead of 1/31/06 if `en-US` was in use).

The following table lists some of the culture names you can use.

Culture Name	Language – Country/Region
`en-AU`	English – Australia
`en-CA`	English – Canada
`en-GB`	English – United Kingdom
`en-NZ`	English – New Zealand
`en-US`	English – United States (this is the default)
`en-ZA`	English – South Africa

Example of changed culture

```
<!--  GLOBALIZATION
       This section sets the globalization settings of the application.
-->
<globalization  culture="en-GB" requestEncoding="iso-8859-1"
   responseEncoding="iso-8859-1"/>
```

In conjunction with setting the culture, you can configure the currency format that displays the total cost in reports to show the applicable currency symbol for that culture. See page 5.

To ensure consistent date formats (meaning those generated by *Genie* and *WebPublisher PRO*) and that date searching performs as expected, you should also specify the equivalent settings in the DBText.INI or InmCSrv.INI file in the *WebPublisher PRO* installation folder. Reset *Genie* after you do this.

For example, for `culture="en-GB"`, the equivalent settings in the [defaults] section of the DBText.INI or InmCSrv.INI file would be:

```
[defaults]
sShortDate=d/M/yyyy
iDate=1
```

For more information about the DBText.INI or InmCSrv.INI file and the `sShortDate` and `iDate` parameters, see the TextWorks online help.

Localization

Overview

Translating Genie involves translating all of the messages, strings, and UI elements. These are located in several places, as of Genie v3.0, SP 3:

- GlobalResource.resx contains all of the messages and UI elements used by the Genie application, excluding user-configurable text.
- The .CONFIG files contain the user-configurable information. See page 116 for more specific information about elements in these files that should be translated.
    - Genie.CONFIG contains miscellaneous parameters, some of which appear in the user interface.
    - GenieFieldMaps.CONFIG contains information about all of the textbase fields, including the label that will identify the field to the user if no specific label is used in one of the CONFIG files in the next bullet.
    - Reports.CONFIG, Queries.CONFIG, and EditScreens.CONFIG contain definitions for reports, query screens, and edit screens. In many cases, they include text for box labels, on-screen hints, and so forth.
    - Labels.CONFIG contains mostly field specifications, but one of the labels defined in there includes text.
    - NavBarMenuData.CONFIG contains the menu text.
    - UserData.CONFIG and Statistics.CONFIG contain a modest amount of text that displays on the MyGenie page and Statistics page, respectively.
- Several HTML e-mail templates contain text that appears in the e-mail, with placeholders for information coming from the textbase records. Example: OverdueEmailBatch.htm.
- Textbase forms used by Genie that include English text or labels should be modified.

Related Applications

Genie uses two other applications that would have to be localized as well.

- InmagicBrowse has messages and UI elements isolated in a resource file, Index_Browse.aspx.resx. It may also have a few user-configurable UI elements in Web.config. If present, these can be translated or removed, as they serve only to override the original button and link captions. (Remove them for multi-lingual Genie, as there can be only one Web.config file.)
- *WebPublisher PRO* (WPP) has a set of messages that it may return to the client browser. These can be translated outside of the software in an editable message file. This message file is requested by appending a parameter to the query sent to WPP: &MF=MyMsgFile.ini (substitute appropriate name). The English message file ships with the product, for ease of customization. Note that that WPP's messages are built into the product; they are not automatically read from this INI file. Only the presence of an MF parameter causes WPP to substitute text from that specified file for its native text.

Multilingual Genie

As described above, a simple translation of Genie involves changing the content of a series of files that are not compiled into the DLL.

To build a multilingual Genie, with support for a set of languages available for selection by the user, perform the same steps as a simple translation but give the files themselves names that incorporate the locale name between periods before the file extension. For example, a French Canadian file set would have names like these:

- GlobalResource.fr-CA.resx
- Genie.fr-CA.CONFIG, Queries.fr-CA.CONFIG, etc.
- OverdueEmailBatch.fr-CA.htm, etc.

Note that Genie's CONFIG files already come in two variants, Xxx.CONFIG and MyXxx.CONFIG. Xxx.CONFIG contains the original configuration shipped with Genie, and MyXxx.CONFIG contains a copy of the same configuration, that can be customized by the end-user and that will not be overwritten by a newer release of Genie. MyXxx.CONFIG is used by Genie if found, otherwise Xxx.CONFIG is used.

If a Genie installation is set up to be multilingual, Genie looks for CONFIG files in this order, stopping as soon as it finds one: MyXxx.yy-ZZ.CONFIG, Xxx.yy-ZZ.CONFIG, MyXxx.CONFIG, Xxx.CONFIG (where Xxx is the original name of the CONFIG file, and yy-ZZ is the selected locale).

Localized help will be sought in a folder named WebHelp.fr-CA (instead of WebHelp). If the folder does not exist or does not contain a file named GenieWebHelp.htm, the help links will point to the WebHelp folder instead.

To enable language selection by end users, populate the <LanguageSetting> element in Genie.CONFIG (all variants). If it is present but commented out, remove the comment indicators. If it is not present, put it in the <configuration> element, perhaps after <appSettings>. Example:

```
  <LanguageSettings SuppressIfSpecified="true">
     <LanguageSetting Name="en-US" Display="English (U.S.)" />
     <LanguageSetting Name="fr-CA" Display="Français (Canada)" MsgFile="CsEngMsg.fr-
CA.ini" />
  </LanguageSettings>
```

Important Notes About Multilingual Genie

- The locale selected by the user affects the language of the user interface only. The locale specified in Web.config is used to determine date recognition and number formatting.
- E-mail templates can be specified in the Statistics.CONFIG file. There is no need to change the template name for a localized Statistics.CONFIG file, as that file name is decorated with the culture name as described above, as well.
- There is English content in several textbases that should not be localized, as the terms in specific fields drives the behavior of the software. These are:
  - The KeyType field in GenieKey must contain one of these values:
    role, **group**, **user**
  - The KeyRoleName, KeyUserRole, and KeyGroupRole fields in Genie Key must contain one of these values:
    Administrators, **Catalogers**, **Staff**, **Public**
  - The ResType field in Reserves must contain one of these values:
    location, **title**, **item**
  - The ResStatus field in Reserves is set by Genie to one of these values:
    active, **canceled**, **closed**
- The software can display system messages returned with exceptions triggered during processing. These are not under Genie's control, and will likely follow the server operating system language.

Customized English Genie

Follow these steps to customize some of the terminology used in the English version of Genie.

- Make a culture-decorated copy of the GlobalResource.resx file, for example: GlobalResource.en-US.resx. Make any desired terminology changes in this copy. For example, change "borrower" to "patron". Make similar changes in culture-decorated copies of CONFIG files, if needed.
- Add or enable the `<LanguageSettings>` entry in MyGenie.CONFIG, listing only your customized culture.
- Reset Genie.

You may also need to set the uiCulture in Web.CONFIG, in the <globalization …> element:

```
<globalization . . . culture="en-NZ" uiCulture="en-NZ" />
```

Genie will use the customized resources and other files, and these files will not be overwritten when you install a newer version of Genie. However, your customized resources won't contain any of the text that may have been added in the newer version. To add this text to your customized files, make a backup of the undecorated GlobalResource.resx file BEFORE you install the upgrade, and then compare it to the GlobalResource.resx file AFTER you install the upgrade. There are some good, free file comparison utilities available on the Web. See, for example, http://en.wikipedia.org/wiki/Comparison_of_file_comparison_tools.

Localizable Information in Genie CONFIG Files

Genie.config

Within the **appSettings** element, the values associated with specific keys contain English text:

- **VolumeIssueFormat**, **MovedIssueFormat**, **MissingIssueFormat** (retain the letters in curly braces)
- **GenieHeading** says "Inmagic Genie", should keep that (end users can modify)

Elsewhere:

- **Display** attributes anywhere
- **CartMessage** attribute of the <**OpacCart**> element
- <**LinkText**> element (2 places), and **XXXLinkText** attributes in several places, including commented-out XML
- <**LoanTerms**> element has subelements with **DisplayTerm** attributes
- <**OrderTypes**>, <**PaymentMethods**>, and <**Currencies**> contain English terms that populate droplists on order edit screens

GenieFieldMaps.config

- <**FieldMap**> **Heading** and **HistoryLink** attributes
- <**Field**> **BoxLabel** attribute

Queries.config

- <**QueryScreen**> **Title** and **BrowseButtonText** attributes as well as Boolean attribute values AND, OR, and NOT
- <**QueryField**> **Display** attributes
- <**QueryField** Type="ButtonRow"> button text attributes
- <**SearchTips**> **Title** and **TipContent** attributes

EditScreens.config

- <**EditScreen**> **TitleXXX** attributes
- <**ToolBar**> **Title** attribute
- <**Button**> **Text** attribute
- <**EditSection**> and <**SubSection**> **Title** attribute
- <**EditTips**> **Title** and **TipContent**
- **NewHelpText**, **XXXLinkText**, and **ToolTip** attributes
- **Display** attributes anywhere

Labels.config

- One label definition, named **LabelRoute3on8.5x11**, contains text in a few **Display** and **Header** attributes of <**Field**> elements

Statistics.config

- <**Label**> elements
- <**Email**> **Subject** and **LabelList** attributes (also, if the **TreatToCImageAs** attribute is "ImageLink(text…)", the link text in parentheses should be translated)

UserData.config

- <**Heading**> elements

NavBarMenuData.config

- <**Item**> **Text** attributes

Reports.config

- <**inmg:Report**> **PrintHeader** and **PrintFooter** attributes
- The following attributes of <**Field**> elements: **Header**, **NoContentHeader**, **ColumnHeader**, **Display**
- The following attribute of <**ItemInfo**>, <**SerialInfo**>, or <**ReserveInfo**> elements: **Heading**
- <**inmg:ReportList**><**Report**> **Display** attribute

End of document