

Defect Details

Defect Id: 44

Project: Gallery Server Pro 2

Name: A ZIP file nested in an uploaded ZIP file is corrupted

Date Found: 9/30/2007

Build #: 2.0.2829

Severity: Low Impact

Date Fixed: 11/2/2007

Build # of Fix: 2.0.2862

Status: Closed

Estimated Work: 4 hrs

Assigned To: Roger Martin

Priority: Low

Actual Work: 2 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

If the ZIP file extension is enabled, and a ZIP file has been prepared that contains one or more ZIP files, and that ZIP file is uploaded with the "Treat ZIP files as media objects and do not extract contents" unchecked, the nested ZIP file is added but when the user attempts to download it, it cannot be opened.

Gallery Server should loop through nested ZIP files and extract its contents.

Replication Procedures

Resolution

Modified ZipUtility class so that if it encounters an embedded ZIP file as it processes a parent ZIP file, it recursively calls itself to extract the embedded ZIP file before moving on.

Notes

Defect Id: 51

Project: Gallery Server Pro 2

Name: Account page throws exception when opened by anonymous user

Date Found: 10/2/2007

Build #: 2.0.2830

Severity: Low Impact

Date Fixed: 11/3/2007

Build # of Fix: 2.0.2863

Status: Closed

Estimated Work: 30 min

Assigned To: Roger Martin

Priority: Medium

Actual Work: 10 min

Reported By: Roger Martin

Percent Complete: 100

Description

The following function is invoked during page load in myaccount.aspx:

```
private void ConfigureControlsFirstTime()
{
    MembershipUser user = Membership.GetUser();
    litUserName.Text = user.UserName;
    txtEmail.Text = user.Email;
}
```

This throws a NullReferenceException when an anonymous user opens the page because the user variable is null.

This page should only be accessible to authenticated users. Move the web page to a folder that requires authentication or add code in the page load event to redirect to the login page if the user is not logged on.

Replication Procedures

Defect Details

Resolution

Added code to page load event to redirect anonymous users to home page. Also made the same change to changepassword.aspx.

Notes

Defect Id: 52

Project: Gallery Server Pro 2

Name: Delete hi-res function incorrectly renames files

Date Found: 10/4/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 10/4/2007

Build # of Fix: 2.0.2833

Status: Closed

Estimated Work: 4 hrs

Assigned To: Roger Martin

Priority: Medium

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

When the delete hi-res function is invoked, Gallery Server replaces the name of the compressed file with the name of the original file. If the compressed file is a different type than the original (say, JPEG versus BMP), then the compressed file is given a name ending in BMP. This causes Gallery Server to later think the image is not a JPEG and refuses to render the image in the browser (it provides a download link only)

Need to update the file name copy routine to preserve the file extension of the compressed image.

Replication Procedures

Resolution

Notes

Defect Id: 53

Project: Gallery Server Pro 2

Name: The HTML container for media object title should be a div, not a p.

Date Found: 10/5/2007

Build #: 2.0.2830

Severity: Low Impact

Date Fixed: 11/7/2007

Build # of Fix: 2.0.2867

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: Medium

Actual Work: 10 min

Reported By: Roger Martin

Percent Complete: 100

Description

When viewing a media object, the title is rendered inside a p tag. If the title contains HTML such as p tags, then this is technically invalid HTML, as p tags cannot be nested inside other p tags.

Change the HTML container tag to a div.

Replication Procedures

Resolution

Changed the p element to a div element.

Notes

Defect Details

Defect Id: 54

Project: Gallery Server Pro 2

Name: Lots of media object title text is covered by Gallery Server logo

Date Found: 10/5/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 11/7/2007

Build # of Fix: 2.0.2867

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: High

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

The tag containing the media object title on mediaobjectview.ascx is hard coded to be 50px high. When there is more text than fits into this, it spills over the borders, which is fine. But the GS logo at the bottom of the page does not reflow to make room for this additional text. End result is that some of the text is covered by the logo.

Replication Procedures

Resolution

Resolved by changing the CSS classes editableContentOff and editableContentOn to use min-height instead of height. IE 6 doesn't understand min-height, so for that browser we are effectively removing the height spec of 50px, so it is now the height of the block element that contains the caption. For large captions it expands as desired. For small, one-line captions, the clickable area is now only about 25px high instead of 50px, but that is acceptable.

Notes

Defect Id: 55

Project: Gallery Server Pro 2

Name: HTML in media object title causes javascript error in IE7

Date Found: 10/5/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 11/7/2007

Build # of Fix: 2.0.2867

Status: Closed

Estimated Work: 8 hrs

Assigned To: Roger Martin

Priority: High

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

IE7 (and possibly other versions of IE) throws a javascript error "A RunTime error has occurred. Do you wish to debug? Line: 1056; Error: Unknown runtime error" whenever the media object title has embedded HTML and the user double clicks it to go into edit mode. Firefox does not have an issue.

Also, when this occurs, the text that appears in the edit box is for the original media object from when the page first loaded, and not for the current one (assuming the user has used the Next/Previous buttons to navigate to a different object.)

Replication Procedures

Resolution

IE doesn't like it when assigning text containing HTML to the innerHTML property of a textarea element. Instead, I changed the assignment to use the value property instead:

```
$get('taCaption').value = $get(_moTitle).innerHTML;
```

Defect Details

This is inside the editCaption javascript function that is generated inside the GetEditMediaObjectScript server function in the mediaobjectview user control code-behind.

Notes

Defect Id: 56

Project: Gallery Server Pro 2

Name: Error when extracting metadata item

Date Found: 10/7/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 10/7/2007

Build # of Fix: 2.0.2836

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: High

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

An ArgumentException is sometimes thrown when extracting metadata from an image. Specifically, a user had an image where one of the metadata items was internally tagged as being ASCII, but the value was null. This caused a problem when I tried to extract the string using this line in function ExtractPropertyValueString (class MetadataItem):

```
return _asciiEncoding.GetString(this._propItem.Value);
```

Replication Procedures

Resolution

I modified the function to check for null as follows:

```
private string ExtractPropertyValueString(ref ExtractedValueType formattedValueType)
{
    formattedValueType = ExtractedValueType.String;

    if (this._propItem.Value == null)
        return String.Empty;

    return _asciiEncoding.GetString(this._propItem.Value);
}
```

Also I modified several other functions to catch exceptions that might occur when extracting other types of data such as integers and longs. Code inside the exception handler substitutes 0 when trying to extract a number and String.Empty when trying to extract a string.

Notes

This was the only bug fix in the GalleryServerPro.Business project since the 2.0.2830 release so I compiled the project and posted to the forum post that initiated this bug:

<http://www.galleryserverpro.com/forum/default.aspx?g=posts&t=60>

Defect Details

Defect Id: 57

Project: Gallery Server Pro 2

Name: Gallery Server does not update the name of the root album when it is edited

Date Found: 10/13/2007

Build #: 2.0.2830

Severity: High Impact

Date Fixed: 11/4/2007

Build # of Fix: 2.0.2864

Status: Closed

Estimated Work: 2 hrs

Assigned To: Roger Martin

Priority: High

Actual Work: 30 min

Reported By: Roger Martin

Percent Complete: 100

Description

When editing the name of the root album, the dialog closes and the cursor changes to an hourglass, but the update never seems to complete. However, when you click the All albums link, you can see that the name was changed. So for some reason, the AJAX callback may not be calling back to the client page correctly.

The problem does not occur for non-root albums.

Replication Procedures

Resolution

Error was caused by an ArgumentException exception thrown from the parameter validation code in HelperFunctions.ValidateDirectoryName during the call to GspPage.UpdateAlbumInfo(). Since root albums do not have a directory name that reflects the album's title, this function should not get called for root albums. Here was the original code in GspPage.UpdateAlbumInfo:

```
if (album.Title != albumEntity.Title)
{
album.Title = WebsiteController.CleanHtmlTags(albumEntity.Title);
album.DirectoryName = HelperFunctions.ValidateDirectoryName(album.Parent.FullPhysicalPath, album.Title);
}
```

Code was changed to this:

```
if (album.Title != albumEntity.Title)
{
album.Title = WebsiteController.CleanHtmlTags(albumEntity.Title);
if (!album.IsRootAlbum)
{
// Root albums do not have a directory name that reflects the album's title, so only update this property for non-root albums.
album.DirectoryName = HelperFunctions.ValidateDirectoryName(album.Parent.FullPhysicalPath, album.Title);
}
}
```

Notes

Defect Details

Defect Id: 58

Project: Gallery Server Pro 2

Name: Incorrect popup help text

Date Found: 10/13/2007

Build #: 2.0.2830

Severity: Low Impact

Date Fixed: 11/7/2007

Build # of Fix: 2.0.2867

Status: Closed

Estimated Work: 10 min

Assigned To: Roger Martin

Priority: Low

Actual Work: 5 min

Reported By: Roger Martin

Percent Complete: 100

Description

The following text appears in the popup help window for the Add album permission on the Manage Roles page:

Note 1: When moving an album, users must have this permission on the destination album.

I think it should say:

Note 1: When moving or copying an album, users must have this permission on the destination album.

Replication Procedures

Resolution

Changed the text as suggested.

Notes

Defect Id: 59

Project: Gallery Server Pro 2

Name: New album created with Create album command does not inherit private setting

Date Found: 10/16/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 11/7/2007

Build # of Fix: 2.0.2867

Status: Closed

Estimated Work: 30 min

Assigned To: Roger Martin

Priority: High

Actual Work: 10 min

Reported By: Roger Martin

Percent Complete: 100

Description

Albums added with the Create album command should inherit the private setting of its parent album, but instead the IsPrivate property defaults to false for all albums. This is related to bug # 40, which fixed the non-inheriting issue for newly added albums during ZIP uploads and synchronizing.

Replication Procedures

Resolution

Added the following line to btnOkClicked() in createalbum.aspx:

```
newAlbum.IsPrivate = parentAlbum.IsPrivate;
```

Notes

Defect Details

Defect Id: 60

Project: Gallery Server Pro 2

Name: Album date start and date end sometimes saved incorrectly

Date Found: 10/17/2007

Build #: 2.0.2830

Severity: High Impact

Date Fixed: 11/7/2007

Build # of Fix: 2.0.2867

Status: Closed

Estimated Work: 2 hrs

Assigned To: Roger Martin

Priority: High

Actual Work: 10 min

Reported By: Roger Martin

Percent Complete: 100

Description

A user in the UK has reported that the date start and date end of an album are always saved one day earlier than specified. For example, when entering 17/10/2007 for Oct 17, 2007, the page later displays 16/10/2007 when it is refreshed.

This may be caused by the time component of the date being saved in the database field, even though time is not relevant in this case. I noticed that dates are saved with 5:00:00 AM as the time component for all start and end dates on my system.

Perhaps the fix is to modify the UpdateAlbumInfo method in GspPage. Change the following:

```
album.DateStart = albumEntity.DateStart;  
album.DateEnd = albumEntity.DateEnd;
```

to this:

```
album.DateStart = albumEntity.DateStart.Date;  
album.DateEnd = albumEntity.DateEnd.Date;
```

Replication Procedures

Resolution

Made the suggested change to use the Date property of the DateTime object.

Notes

Defect Id: 61

Project: Gallery Server Pro 2

Name: WebSiteController.GetRolesForUser not thread safe

Date Found: 10/21/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 11/8/2007

Build # of Fix: 2.0.2868

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: High

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

Need to lock the dictionary in WebSiteController.GetRolesForUser when adding the roles for a user. Inside the lock, check for the existence of the key and only add it if not present.

Defect Details

Error from GS demo site 2007-10-21:

Name

Replication Procedures

Resolution

In WebSiteController.GetRolesForUser, I modified the following line:

```
rolesCache.Add(HttpContext.Current.Session.SessionID, roles);
```

to this:

```
lock (rolesCache)
{
if (!rolesCache.ContainsKey(HttpContext.Current.Session.SessionID))
{
rolesCache.Add(HttpContext.Current.Session.SessionID, roles);
}
}
```

Notes

Defect Id: 62

Project: Gallery Server Pro 2

Name: Url is incorrectly created after rotation

Date Found: 10/29/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 11/7/2007

Build # of Fix: 2.0.2867

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: High

Actual Work: 1.5 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

The url is incorrectly created after rotating a media object in certain cases. Steps to repro:

1. Navigate to compressed version of image.
2. Click rotate toolbar icon. Rotate image.
3. You are redirected back to the media object view page. Now click Next or Previous to go to another image.
4. Click the rotate link again. Rotate image.
5. The URL you are redirected to is incorrect. It appears like this:

<http://www.galleryserverpro.com/gsdemo/default.aspx&moid=3254?msg=30>

Replication Procedures

Resolution

This was caused by a bug in this code in GspPage.GetReferringPageUrl:

```
System.Text.RegularExpressions.Regex rgx = new System.Text.RegularExpressions.Regex("[?&]msg=[0-9]+",
System.Text.RegularExpressions.RegexOptions.IgnoreCase);
url = rgx.Replace(url, "");
```

Defect Details

It is supposed to remove the "msg" query string parameter, but it caused urls like `www.mysite.com/default.aspx?msg=10&moid=100` to become `www.mysite.com/default.aspx&moid=100`, which is incorrect. I replaced the above lines with:

```
url = WebsiteController.RemoveQueryStringParameter(url, "msg");
```

This was a helper function I had previously created and I should have been using it here also. It parses the query string and correctly removes the "msg" query string parm without corrupting the url.

Notes

Defect Id: 63

Project: Gallery Server Pro 2

Name: Error: Gallery Server Pro cannot write to the media object directory

Date Found: 10/30/2007

Build #: 2.0.2830

Severity: High Impact

Date Fixed: 10/30/2007

Build # of Fix: 2.0.2859

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: High

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

After posting the CodeProject article about Gallery Server Pro, the demo site failed between 3 and 4 in the morning for each of the next two nights. There were two errors that occurred back to back:

Error 1:

Type

Replication Procedures

Resolution

The bug was fixed in two parts:

1. Added a lock around the code in `HelperFunctions.ValidatePhysicalPathExistsAndIsReadWritable` that validates the `test.config` filename and creates the file. This should prevent multiple threads from trying to access the same file.
2. More importantly, the property setters in the `AppSetting` class were modified in these properties: `MediaObjectPhysicalPath`, `ThumbnailPath` & `OptimizedPath`. Previously, each setter would call `HelperFunctions.ValidatePhysicalPathExistsAndIsReadWritable(value)` even if it was assigning the same path that had been previously checked for one of the other properties. I added a class-level `StringDictionary` member variable to hold the list of paths that have been checked, and I only call the `Validate...` method when encountering a new path that has not been previously checked. For example, this will occur when thumbnails are stored in a different directory than the media objects.

Notes

Defect Details

Defect Id: 65

Project: Gallery Server Pro 2

Name: Error - Violation of PRIMARY KEY constraint 'PK_gs_Synchronize'

Date Found: 10/31/2007

Build #: 2.0.2830

Severity: Low Impact

Date Fixed: 11/8/2007

Build # of Fix: 2.0.2868

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: Medium

Actual Work: 2 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

The following error occurred on 10-31-2007:

Error - Violation of PRIMARY KEY constraint 'PK_gs_Synchronize'. Cannot insert duplicate key in object 'dbo.gs_Synchronize'. The statement has been terminated.

Replication Procedures

Resolution

This was caused by not taking proper precautions to prevent concurrency issues. While this particular bug was isolated to the stored procedure `gs_VerifyMinimumRecords`, this type of bug affected two others - `gs_RoleUpdate` and `gs_SynchronizeSave`. I fixed them all by wrapping certain sections in an explicit transaction, added the SQL `'WITH (UPDLOCK, HOLDLOCK)'` to certain statements, and set the option `'SET NOCOUNT, XACT_ABORT ON'` at the beginning of each procedure.

An excellent blog post explaining the issue is at <http://weblogs.sqlteam.com/dang/archive/2007/10/28/Conditional-INSERTUPDATE-Race-Condition.aspx>

For example, here was an original piece of SQL from `gs_VerifyMinimumRecords`:

```
IF NOT EXISTS (SELECT * FROM [gs_Gallery] WHERE GalleryID = @GalleryID)
BEGIN
  INSERT gs_Gallery (GalleryId, Description)
  VALUES (@GalleryId, 'My Gallery')
END
```

It was modified to this:

```
BEGIN TRAN
IF NOT EXISTS (SELECT * FROM [gs_Gallery] WITH (UPDLOCK, HOLDLOCK) WHERE GalleryID =
@GalleryID)
BEGIN
  INSERT gs_Gallery (GalleryId, Description)
  VALUES (@GalleryId, 'My Gallery')
END
COMMIT
```

Notes

Defect Details

Defect Id: 66

Project: Gallery Server Pro 2

Name: Error - An item with the same key has already been added.

Date Found: 11/1/2007

Build #: 2.0.2830

Severity: Low Impact

Date Fixed: 11/9/2007

Build # of Fix: 2.0.2869

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: Medium

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

The error below has occurred in the GS demo site. It is probably caused by multiple threads accessing the same section of code that deals with the cached media objects. Investigate using the lock statement around critical sections to make it thread-safe.

Error - An item with the same key has already been added.

Replication Procedures

Resolution

The code to add a media object to the cache dictionary was not thread-safe. Instead of this in Factory.RetrieveMediaObject:

```
mediaObjectCache.Add(mediaObjectId, mediaObject);  
HelperFunctions.CacheManager.Add(CacheItem.MediaObjects.ToString(), mediaObjectCache);
```

I changed it to this:

```
lock (mediaObjectCache)  
{  
if (!mediaObjectCache.ContainsKey(mediaObjectId))  
{  
mediaObjectCache.Add(mediaObjectId, mediaObject);  
HelperFunctions.CacheManager.Add(CacheItem.MediaObjects.ToString(), mediaObjectCache);  
}  
}
```

I made a similar change to Factory.RetrieveAlbum for the album cache.

Notes

Defect Id: 67

Project: Gallery Server Pro 2

Name: Incorrect help text on Site Settings - General page

Date Found: 11/2/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 11/7/2007

Build # of Fix: 2.0.2867

Status: Closed

Estimated Work: 30 min

Assigned To: Roger Martin

Priority: Medium

Actual Work: 10 min

Reported By: Roger Martin

Percent Complete: 100

Defect Details

Description

The following help text appears for the **Web site heading text URL** property:

The URL the user will be directed to when she clicks any page's header text. Optional: if not present, no link will be rendered. This text appears in the browser's title bar.Examples:

`http://www.mysite.com`

`/` (the root of the web site)

`~/` (the root of the current application)

The phrase "This text appears in the browser's title bar." does not belong with this property. Instead, it should be moved to the **Web site title** property.

Replication Procedures

Resolution

Updated `Cfg_pageHeaderTextUrl_Bdy` and `Cfg_websiteTitle_Bdy` resource strings as suggested.

Notes

Defect Id: 68

Project: Gallery Server Pro 2

Name: Incomplete user message when one or more files are skipped during upload

Date Found: 11/7/2007

Build #: 2.0.2830

Severity: Low Impact

Date Fixed: 11/7/2007

Build # of Fix: 2.0.2867

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: Medium

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

If a user uploads one ore more files that are not allowed because their file extensions are not enabled, the user is given the message "Some media objects were not added". However, the code is designed to provide additional information about *which* files were not added and why. This additional code is not executing.

Replication Procedures

Resolution

The list of skipped files is temporarily stored in a session variable so that the receiving page can access it to display in the error message. The page that stores this list (`addobjects.aspx`) was storing it as a `System.Collections.Specialized.StringCollection` object, but the receiving page (`default.aspx`) was trying to receive it with this code:

```
System.Collections.Generic.List<String> skippedFiles = Session
[GlobalConstants.SkippedFilesDuringUploadSessionKey] as System.Collections.Generic.List<String>;
```

```
detailMsg = string.Empty;
if (skippedFiles != null)
```

```
{
```

```
// This message is unique in that we need to choose one of two detail messages from the resource file. One is for
when a single
```

```
// file has been skipped; the other is when multiple files have been skipped.
```

Defect Details

```
if (skippedFiles.Count == 1)
{
    detailMsg = Resources.GalleryServerPro.ResourceManager.GetString(String.Concat(resourcePrefix,
    this.Message.ToString(), "Single", detailSuffix));
    detailMsg = String.Format(CultureInfo.CurrentCulture, detailMsg, skippedFiles[0]);
}
else if (skippedFiles.Count > 1)
{
    detailMsg = Resources.GalleryServerPro.ResourceManager.GetString(String.Concat(resourcePrefix,
    this.Message.ToString(), "Multiple", detailSuffix));
    detailMsg = String.Format(CultureInfo.CurrentCulture, detailMsg, skippedFiles.Count, ConvertListToHtmlBullets
    (skippedFiles));
}
}
```

As you can see, it was trying to convert the session variable to `System.Collections.Generic.List<String>`. Since the conversion fails, the variable `skippedFiles` ended up being null, which meant the code to generate the `detailMsg` never ran.

The fix was to modify the this code so that the session variable was converted to a `StringCollection`.

Notes

Defect Id: 69

Project: Gallery Server Pro 2

Name: InvalidOperationException while iterating through browser IDs in Config layer

Date Found: 11/8/2007

Build #: 2.0.2830

Severity: Low Impact

Date Fixed: 11/8/2007

Build # of Fix: 2.0.2868

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: Medium

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

The error below sometimes occurs as the code is iterating through browser IDs in `GalleryServerPro.Configuration.BrowserCollection.FindClosestMatchById()`. I believe this is occurring because another thread is modifying the contents of the `ArrayList` while the current thread is iterating. The only code I know of that would modify this is the `Reverse` function that also occurs in the method. Need to make this operation thread safe.

Error - Collection was modified; enumeration operation may not execute.

Replication Procedures

Resolution

Added **lock (browserIds.SyncRoot)** around the code that iterates through the `ArrayList` items in `GalleryServerPro.Configuration.BrowserCollection.FindClosestMatchById()`.

Notes

Defect Details

Defect Id: 70

Project: Gallery Server Pro 2

Name: Incorrect "not matching password" text when changing PW for user on Manage Users page

Date Found: 11/16/2007

Build #: 2.0.2830

Severity: Low Impact

Date Fixed: 11/16/2007

Build # of Fix: 2.0.2876

Status: Closed

Estimated Work: 10 min

Assigned To: Roger Martin

Priority: Low

Actual Work: 10 min

Reported By: Roger Martin

Percent Complete: 100

Description

On the manage users page (manageusers.aspx), the error message that is displayed when an admin attempts to change a user's password but the two passwords don't match is defined like this:

```
<asp:Label ID="lblNotMatchingPasswords" runat="server" Visible="false" ForeColor="red"
Text="* <%$ Resources:GalleryServerPro, Admin_Manage_Users_Passwords_Not_Matching_Error %
*" />
```

However, ASP.NET does not extract the resource. Instead, when it is displayed, it renders it literally, like this:

```
* <%$ Resources:GalleryServerPro, Admin_Manage_Users_Passwords_Not_Matching_Error %>
```

Need to modify it so that the correct message is displayed.

Replication Procedures

Resolution

This was fixed by refactoring the asterisk into the resource file and redefining the label as follows:

```
<asp:Label ID="lblNotMatchingPasswords" runat="server" Visible="false" ForeColor="red"
Text="<%$ Resources:GalleryServerPro, Admin_Manage_Users_Passwords_Not_Matching_Error %>" />
```

Notes

Defect Id: 71

Project: Gallery Server Pro 2

Name: Installer lists Windows Authentication instead of SQL Server Authentication

Date Found: 11/20/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 11/20/2007

Build # of Fix: 2.0.2880

Status: Closed

Estimated Work: 5 min

Assigned To: Roger Martin

Priority: High

Actual Work: 5 min

Reported By: Roger Martin

Percent Complete: 100

Description

The installer screen "SQL Server Connection - Setup", lists Windows Authentication twice. The second item should say SQL Server Authentication. This was a typo.

Replication Procedures

Resolution

Defect Details

Notes

Defect Id: 72

Project: Gallery Server Pro 2

Name: Synchronize needlessly updates original file's creation date/time stamp

Date Found: 11/26/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 11/26/2007

Build # of Fix: 2.0.2886

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: Medium

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

When synchronizing, the function HelperFunctions.GetHashKeyUnique was called for every existing media object within the scope of the synchronize. This function checks the list of existing hash keys and, if one is present that matches the hash key for the current file, updates the creation date/time stamp of the file so that the generated hash key is unique. This function did not notice that the existing hash key it found was for the media object it was currently processing, so it would needlessly update the timestamp. The end result is that new hash keys are generated for every existing media object and the creation date/time stamps are updated to the current date/time.

Replication Procedures

Resolution

The function UpdateExistingMediaObject in SynchronizationManager was updated so that the following line:

```
mediaObject.Hashkey = HelperFunctions.GetHashKeyUnique(mediaObject.Original.FileInfo);
```

was replaced with this:

```
if (mediaObject.Hashkey != HelperFunctions.GetHashKey(mediaObject.Original.FileInfo))
{
    mediaObject.Hashkey = HelperFunctions.GetHashKeyUnique(mediaObject.Original.FileInfo);
}
```

Thus, the hash key is only updated when it doesn't match the one currently assigned to the media object. This will probably only happen when a media object file's creation date/time stamp has been changed since the last synch. I expect this to be an uncommon occurrence. As a result of this change, the hash keys should be largely static once assigned, and the creation date/time stamp is unlikely to be modified.

Notes

Defect Id: 73

Project: Gallery Server Pro 2

Name: Retrieving IsPrivate property on album does not inflate object when necessary

Date Found: 11/26/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 11/26/2007

Build # of Fix: 2.0.2886

Status: Closed

Estimated Work: 2 hrs

Assigned To: Roger Martin

Priority: High

Actual Work: 2 hrs

Reported By: Roger Martin

Percent Complete: 100

Defect Details

Description

In certain circumstances it is possible to access the IsPrivate property of an Album object without the object being inflated. Since the property getter does not verify the object is inflated, it simply returns the default value of the property, which is false.

Add logic to the property getter to ensure the object is inflated. This will guarantee the IsPrivate property is set to the proper value.

Replication Procedures

Resolution

The IsPrivate property is implemented in the GalleryObject class, so the following method call was added to the property getter:

```
VerifyObjectIsInflated();
```

This method did not exist, so it was created:

```
private void VerifyObjectIsInflated()
{
// If this is a pre-existing object (i.e. one that exists in the data store), and it has not been inflated
// from the database, go to the database and retrieve the info for this object.
if (!(this.IsNew) && (!this.IsInflated))
{
this.Inflate();
}
}
```

Notes

Defect Id: 74

Project: Gallery Server Pro 2

Name: Media objects in private albums appear in search results for anonymous users

Date Found: 11/26/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 11/26/2007

Build # of Fix: 2.0.2886

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: High

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

Anonymous users should never see private albums or the media objects within them. The search algorithm uses the IsPrivate property of the media object to determine whether or not it is returned in the result set. However, the IsPrivate property of media objects are always false and never set to true, even when its containing album is private. This causes media objects in private albums to be returned in search results. Clicking on one of these media objects correctly prevents the user from viewing it, but it should not be shown in the first place.

Replication Procedures

Resolution

The algorithm was changed so that when setting an album's IsPrivate setting via the UI, the IsPrivate property of all child media objects are updated in addition to the album and its child albums. The search algorithm then has correct

Defect Details

information when determining whether or not to include a media object in the search results.

Notes

Defect Id: 75

Project: Gallery Server Pro 2

Name: Moving an album to an album that already contains an identically named album causes an exception

Date Found: 11/26/2007

Build #: 2.0.2830

Severity: Medium Impact

Date Fixed: 11/26/2007

Build # of Fix: 2.0.2886

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: High

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

No validation is performed when moving an album to a new parent album. If the parent album already contains a directory with the same name as the album we're moving, an exception occurs.

Replication Procedures

Resolution

Added code in AlbumSaveBehavior.PersistToFileSystemStore to make sure the directory name is valid before attempting to move it. If there is a naming conflict, we slightly alter the name until it is unique.

Notes