**Document Importer For SharePoint
Product Documentation**

Ref	DIFS
Date	23/02/2013
Version	1.0.1.0

Document Control Sheet

Revision History			
Date	Change	Version	
12/06/2012	First release	1.0.0.1	
23/02/2013	Updated to support office 365 and tested with SharePoint 2013	1.0.1.0	

Referenced Sources

Table of Contents

1   Description

Quickly import large volumes of documents into SharePoint document libraries using an effective user interface.
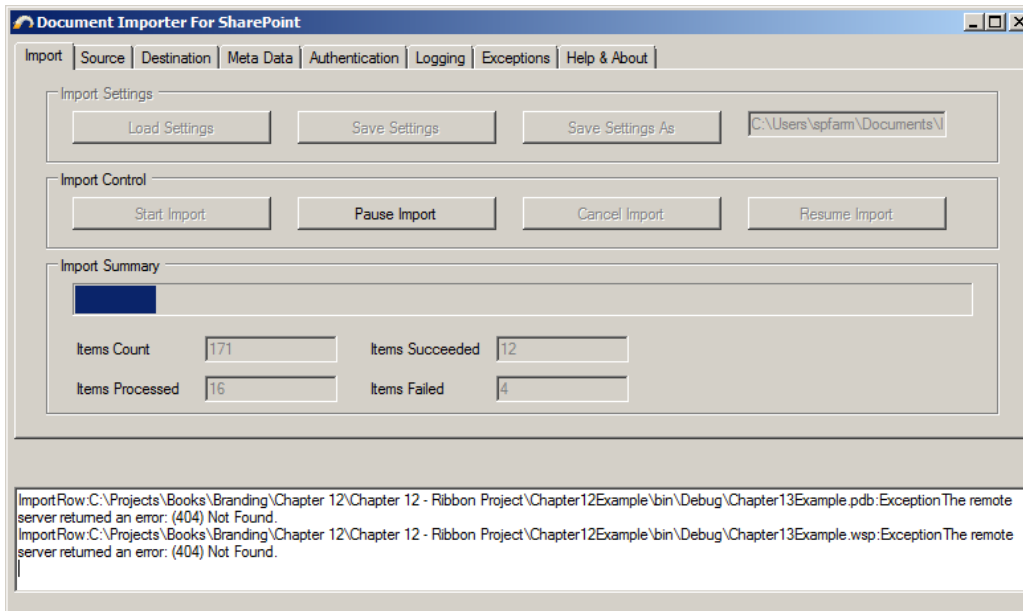
2   Key Features

Key Features	
SharePoint Versions	SharePoint Server 2010 SharePoint Foundation 2010 SharePoint Server 2013 SharePoint Foundation 2013 Office 365 / SharePoint Online
Authentication	Support authentication under the current user's or specified credentials. Supports forms based authentication. Supports Office 365 / SharePoint Online authentication
Meta data	Allows existing file meta data to be imported
Import Control	Imports can be paused, resumed and cancelled. Import progress is reported to the use interface.
Exception handling	Exceptions can be saved for correction and reprocessing.
Save settings	Import settings can be saved and retained for future use.
Uses Client Object Model	You may run the software on the server or a client PC.

3   Installing the software

> If you are installing the software on to a machine other than a SharePoint server (i.e. a client pc) you must first install the 32bit or 64bit (according to that machines o/s) version of the SharePoint Client Object Model.

http://www.microsoft.com/en-us/download/details.aspx?id=21786

If you want to access Office 365 / SharePoint then you also need to install the appropriate version of Windows Identity Foundation for your machine.

http://www.microsoft.com/en-us/download/details.aspx?id=17331

Once pre-requisites have been installed you can install and run the Document Importer for SharePoint (DIFS).

DIFSInstaller.msi

4 Using the software

4.1 *Walkthrough of an import*

The software is intended to import files from an external source into a SharePoint document library.

For this example we are going to walk through importing a local folder on the c:\ drive into SharePoint.

We have deliberately selected an awkward source that includes some problematic files in order to demonstrate the exception handling. If you read the walk through until the end you should then be able to conduct your own imports easily using the software.

First we will choose the source;
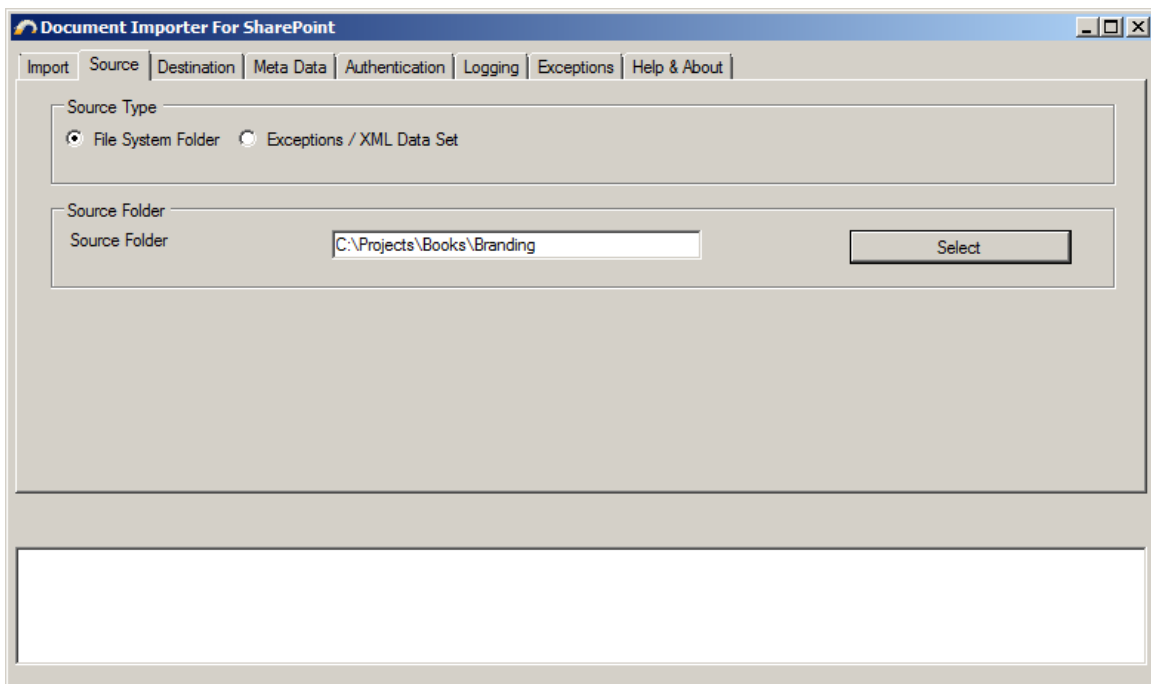
Now we will choose the authentication to use for SharePoint.

Now we will choose the destination;

To select the destination we could have entered the three destination URL's manually. Instead we entered http://intranet and clicked [Load]. This then gave us the list of available sites and libraries to import to.
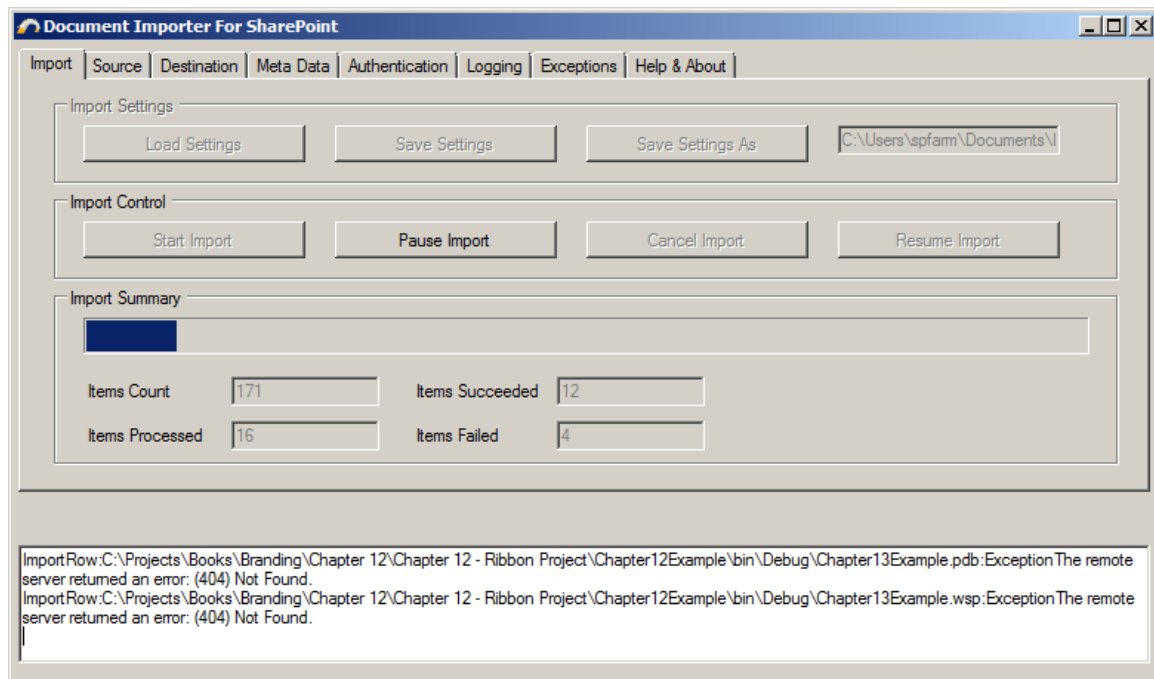
Now we will choose how to set the meta data. In this example we are going to take the document created and modified date from the file system instead of using the current date.
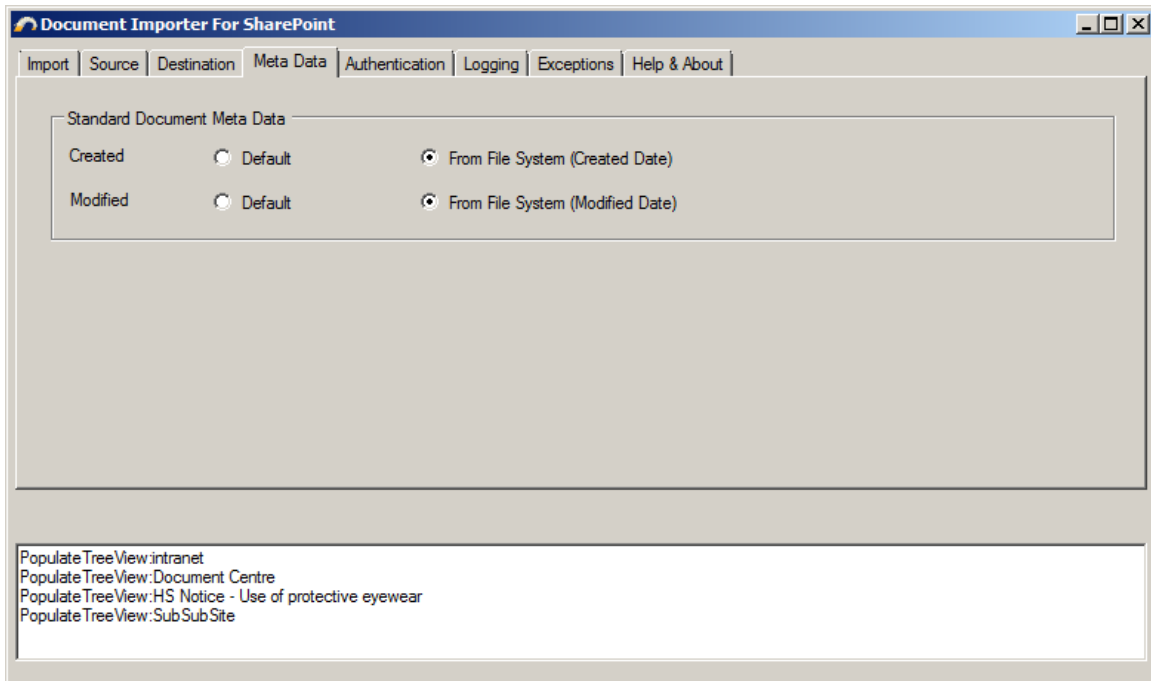
Now we will save our settings. From the import tab we click save settings and enter the name of the settings xml file.
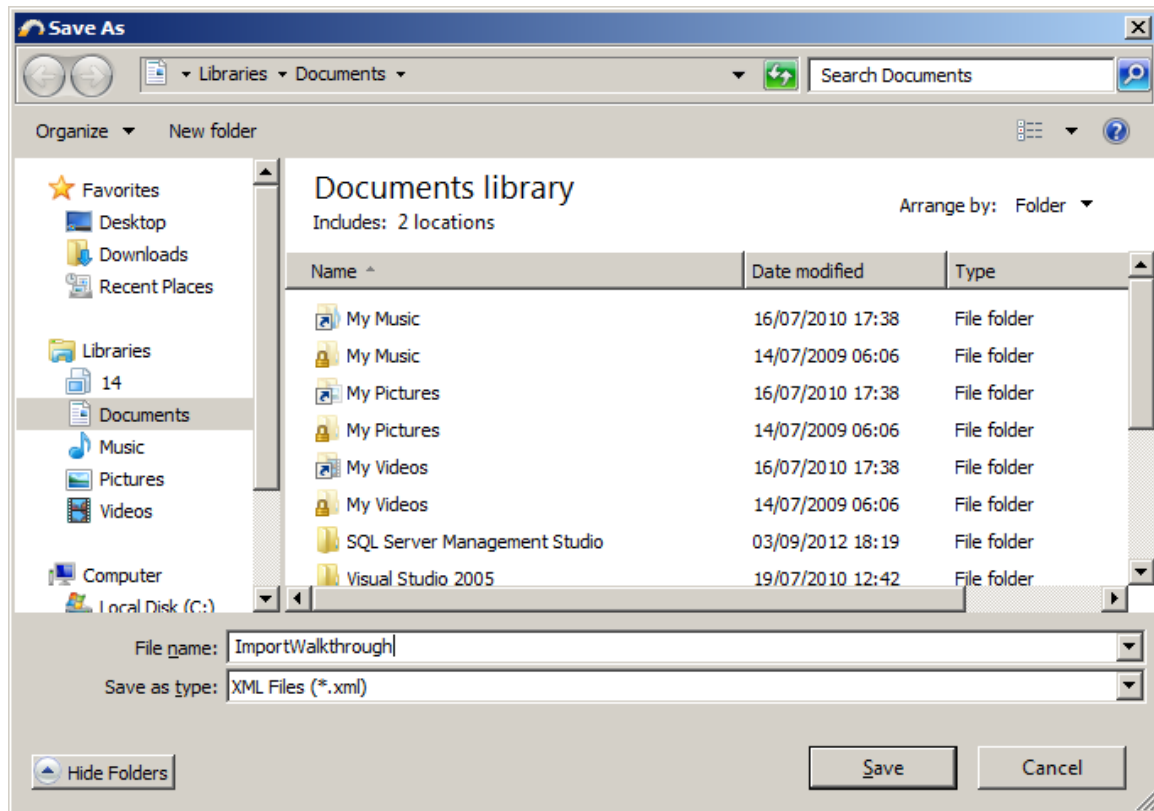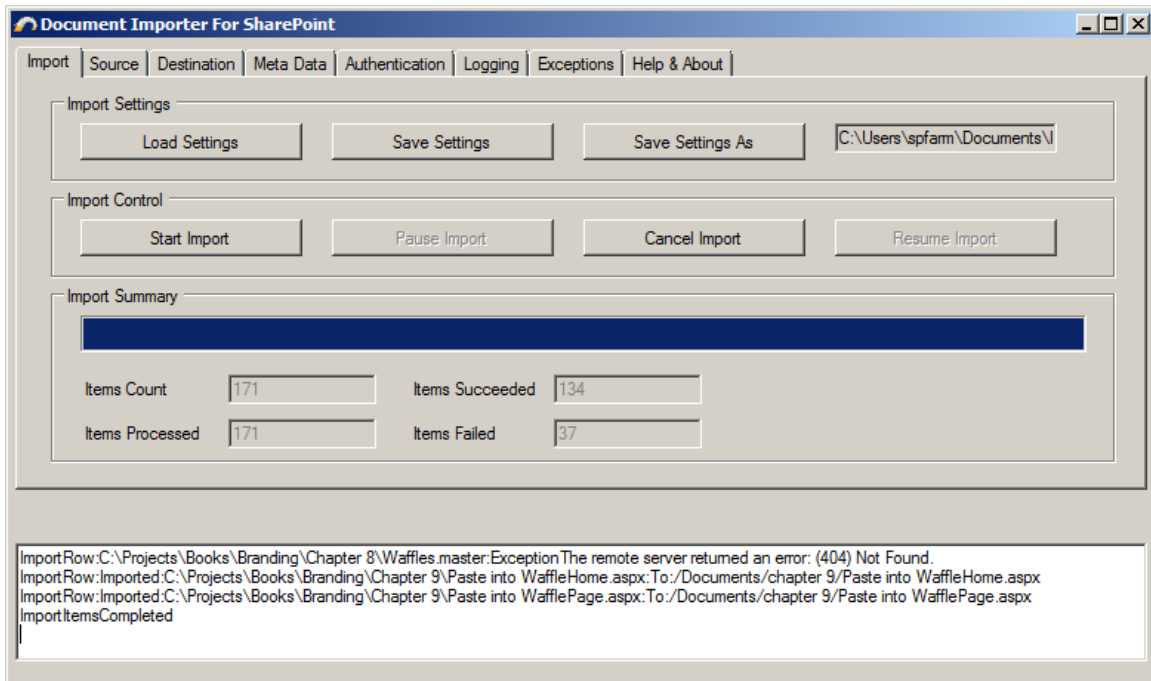
Now we are ready to run our import by clicking [Start Import].

As the import runs it reports progress.

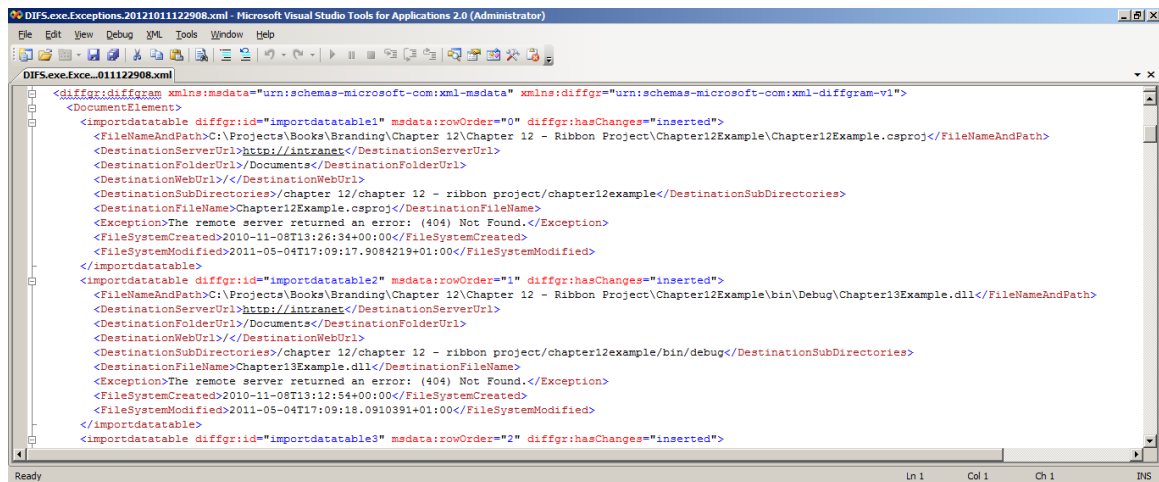You can pause and resume the import as required.

When the import finishes we can see that there were a number of failure items -
37 to be precise.

We can save the details of these from the exceptions tab.

This will save an XML file into the application directory.

We can then close down the importer and open this file up.

In the file we can see an entry for every file which failed the import process.

The XML includes the exception detail as reported by SharePoint (which is not always as helpful as it might be).

In this example the SharePoint error is unclear but when you look at the types of file which have failed they are all extensions which, by default, are not allowed in SharePoint.

This is configured here in central administration;
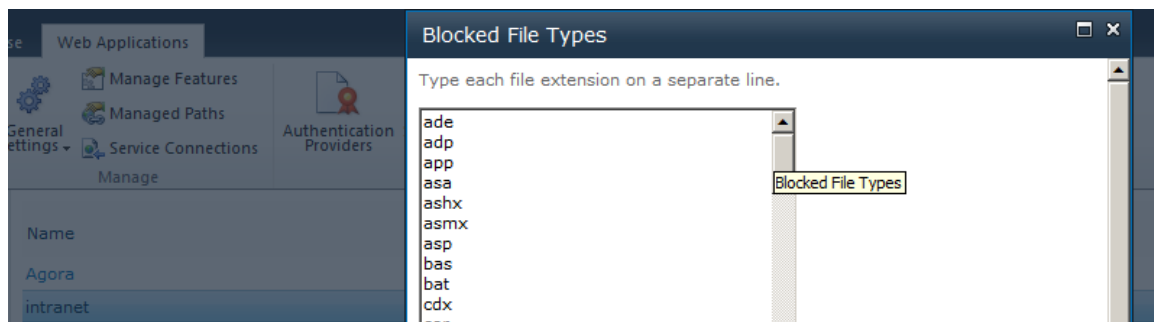
We will, however, not change this configuration of SharePoint which is there for good reason. Instead we will just change the actual file name.

So we edit each entry in the XML;

```
<DestinationFileName>Chapter13Example.dll</DestinationFileName>
```

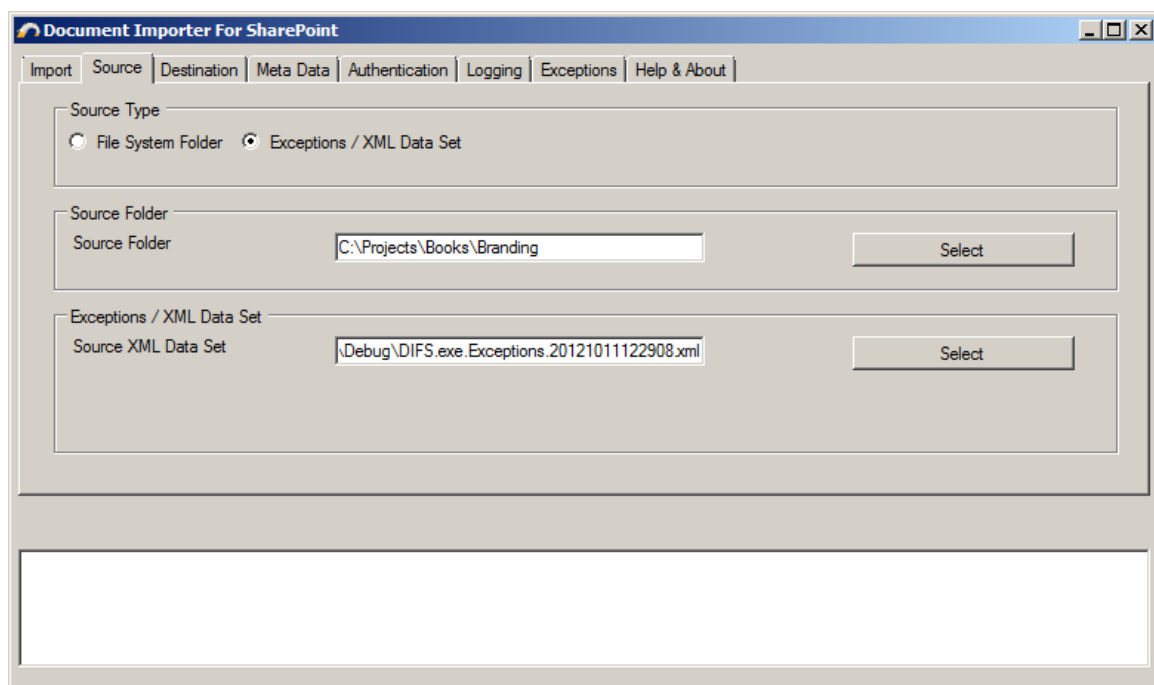Becomes

```
<DestinationFileName>Chapter13Example.dll.allow</DestinationFileName>
```

Now we can load up the importer again.

If we click [Load Settings] and load up the previous settings it will restore all the configuration we made for the first run.

We will then make one change;

Here we have changed the import so that instead of trying to import all the documents again it is just importing all the documents which failed, but with the corrections we have just made.

We will click [Save Settings As] under the [import] tab and save these settings as ImportWalkthroughExceptions.XML just in case we need to do any more exception processing.
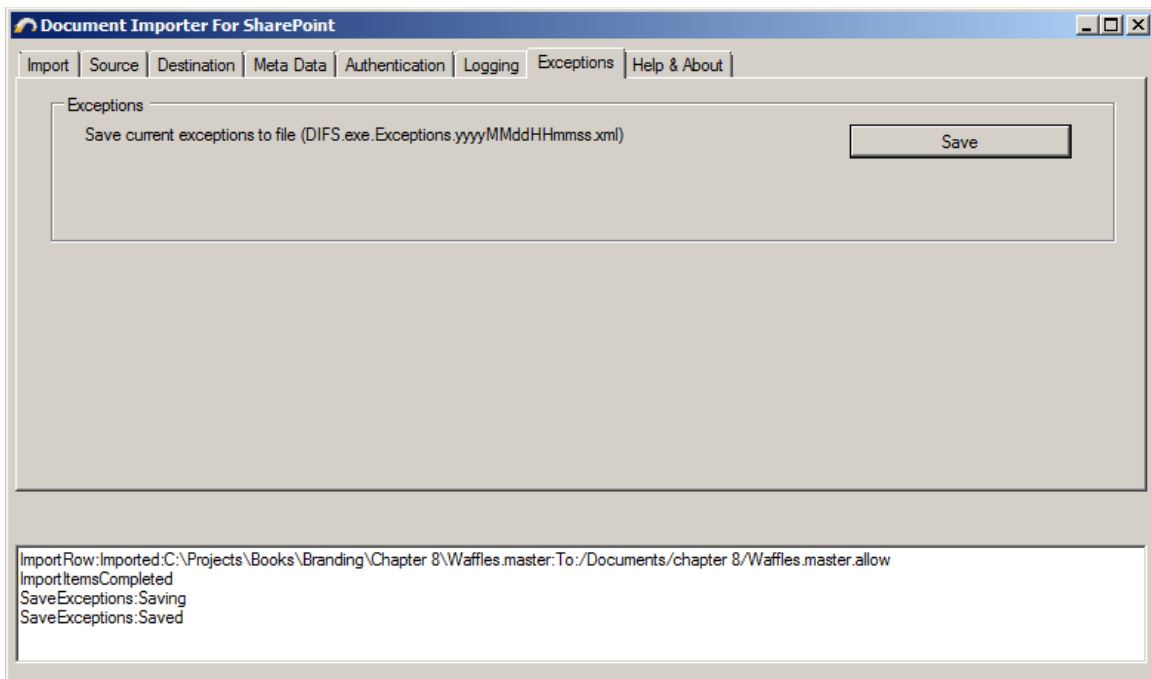
Now we can click [Start Import] to process the exceptions.

The import will run and hopefully it will now import some of the exceptions.
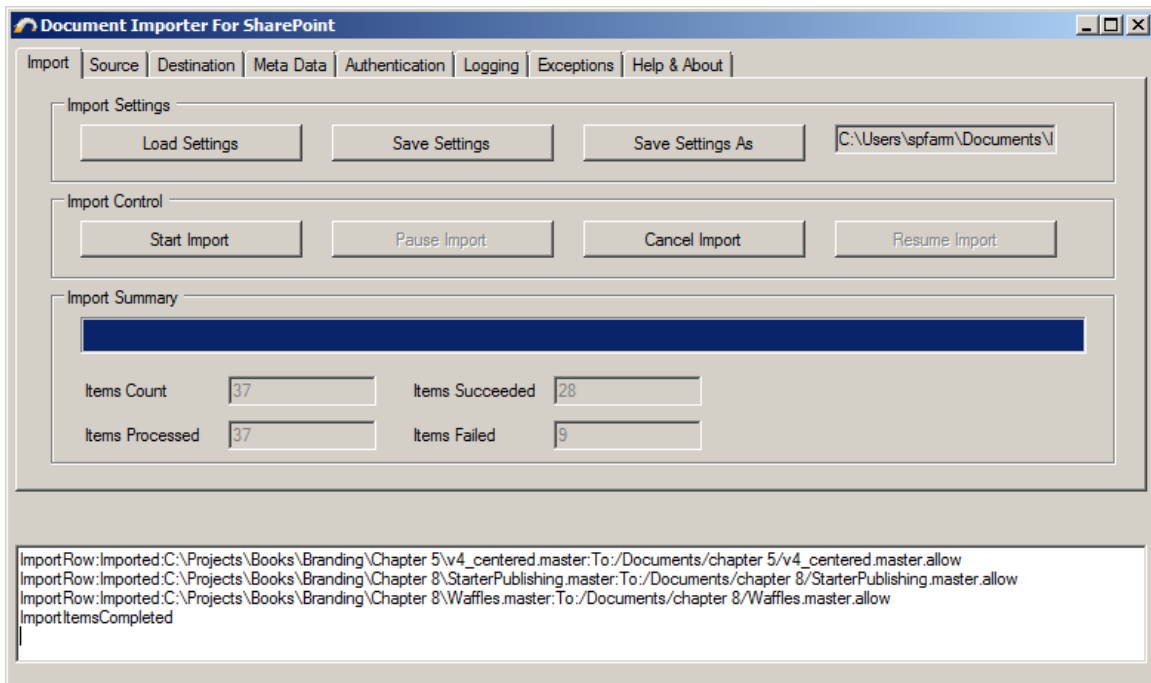
But wait a minute. We still have 9 failed items. This means that 9 of the corrections we made in the XML file were incorrect.

So we will go to the exceptions tab and click [Save Exceptions]

We can now look at that XML file.

```xml
<importdatatable diffgr:id="importdatatable2" msdata:rowOrder="1"
diffgr:hasChanges="inserted">

<FileNameAndPath>..le\bin\Debug\Chapter13Example.pdb</FileNameAndPath>
<DestinationServerUrl>http://intranet</DestinationServerUrl>
<DestinationFolderUrl>/Documents</DestinationFolderUrl>
<DestinationWebUrl>/</DestinationWebUrl>
<DestinationSubDirectories>../bin/debug</DestinationSubDirectories>

<DestinationFileName>Chapter13Example.allow</DestinationFileName>
<Exception> (404) Not Found.</Exception>
<FileSystemCreated>2010-11-08T13:12:56+00:00</FileSystemCreated>
<FileSystemModified>2011-05-
04T17:09:18.1232656+01:00</FileSystemModified>
</importdatatable>
```

So why is this not importing?

Well the problem is acutally in the fact that SharePoint will not allow a directory called "Bin" to be specified in a URL.

So we change

```xml
<DestinationSubDirectories>../bin/debug</DestinationSubDirectories>
```

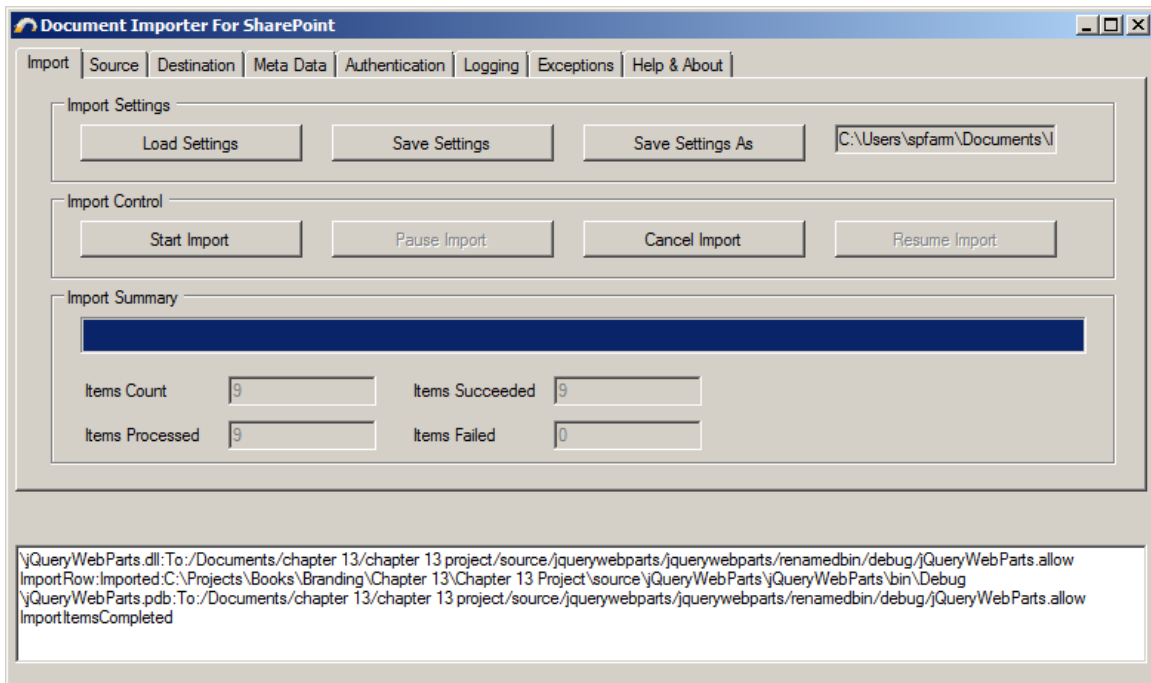To

```xml
<DestinationSubDirectories>../renamedbin/debug</DestinationSubDirectorie
s>
```

For each affected entry in the xml file.

NB: Where we have .. above this is simply indicating that we have truncated the full directory name to make this document more readable.

Now we can re-run the import using the amended XML file as the source and it runs to a successful completion.

And if we look in SharePoint we have all of our directories and their contained files imported.
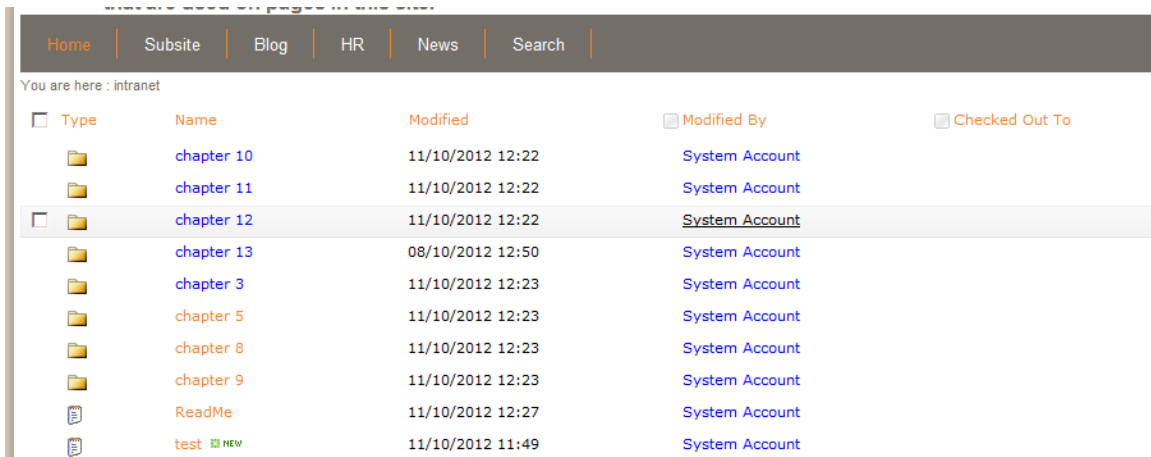
4.2   *Importing To SharePoint Online / Office 365*

You may import files to these platforms.

To do so you must ensure that under authentication you specify [x] Office 365 and your credentials to access the service. Once you have done this you may use the destination tab to select the appropriate library and run and import as you would for a normal site.
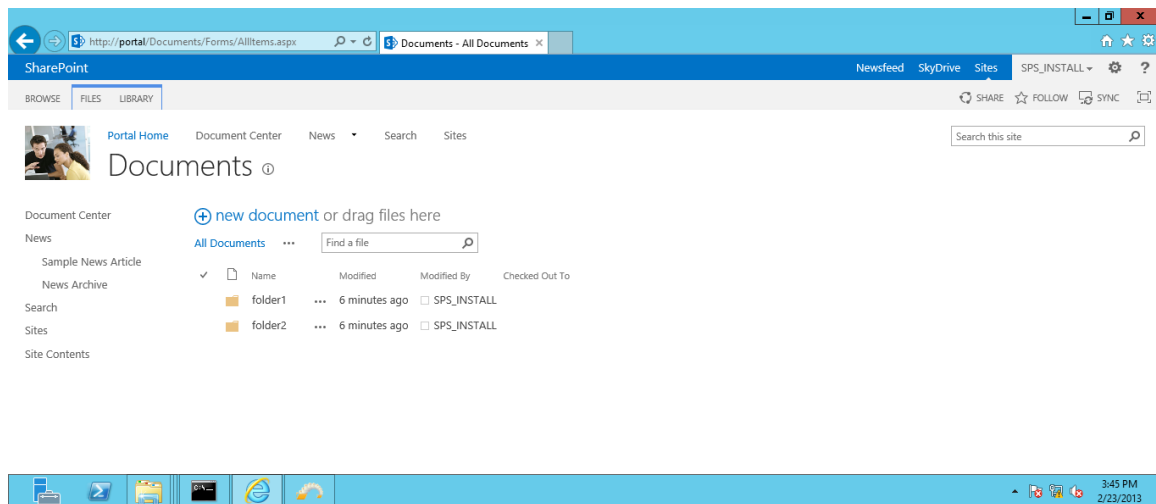
4.3 *Importing To SharePoint 2013*

The application will import to SharePoint 2013 exactly as it does to SharePoint 2010, creating folders if required as shown below.

5   Troubleshooting

Troubleshooting	
Error	Corrective check or action
Could not load file or assembly 'Microsoft.IdentityModel, Version=3.5.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35' or one of its dependencies. The system cannot find the file specified.	This means that Windows Identity Foundation is not installed on the machine. You can download the and install windows update from this location: http://www.microsoft.com/en-us/download/details.aspx?id=17331