HOWTO: Migrate a WingScan based app to 10.5.2 (and newer - includes 10.6, 10.7, 11.0 and 11.1)

Deprecation Notice (Applicability)

This is an older article and references migrating from pre-10.5.2

Those versions have been out of support for a very long time, and with changes in the Chromium engine (powering Google Chrome and MS Edge browsers), it's near impossible that anyone could have a still-working deployment of WebCapture from pre-10.5.2, but this article is here just in case you had some ancient code you need to resurrect.

Main Article (Preserved)

The 10.5.2 version of our SDK includes an entirely rearchitected WingScan.

This change was made because Chrome and FireFox now require Non-NPAPI plugins starting January 2015.

If you would like a full history of Web Scanning as it evolved at Atalasoft, please see

INFO: A Brief History of Web Based Scanning at Atalasoft

No Direct client-side API Changes

The good news is that our new WingScan does not change in terms of the API for initialization, setting options, scanning, and general handling of events/errors.

The only code changes would be if you were handling "noplugin" errors gracefully by detecting "this user is using FireFox/Chrome/IE, but doesn't have the plugin, give them a link to it and a message to install" then you will need to give them a link to the Kofax.WebCapture.Installer.msi in your WebCapture Resources directory (see below)

How Do I Know if I Have the "NEW" or "OLD" Version

The updated WingScan was added in 10.5.2. If your version of DotImage is 10.5.2 or newer, then you have the "new" WingScan.

HOWTO: Migrate a WingScan based app to 10.5.2 (and newer - includes 10.6, 10.7, 11.0 and 11.1)

My WebCapture Resources Folder has Both OLD and NEW WingScan Files

Our initial 10.5.2 installer did not completely remove the old files in the WebCapture Resources directory:

C:\Program Files (x86)\Atalasoft\DotImage 10.5\bin\2.0\x86\WebResources\WebCapture\

So, if you had a 10.5 version prior to 10.5.2, you may notice that you have a mix of old and new WingScan.

You can address this by simply opening

c:\Program Files (x86)\Atalasoft\DotImage 10.5\bin\2.0\x86\WebResources\WebCapture\

in windows explorer, and ensuring that you ONLY have the following 4 files (3 for 10.6 and newer):

```
talaCaptureService.js (this file is not present in 10.6 and forward) atalaWebCapture.js
jquery-1.7.1.min.js Kofax.WebCapture.Installer.msi
```

Here is a quick script that will remove the unneeded files

```
D "c:\Program Files (x86)\Atalasoft\DotImage 10.5\bin\2.0\x86\WebResources\WebCapture\" DEL
atalaEzTwain.js DEL atalaLeader.js DEL atalaUpload.js DEL atalaWebCapture.min.js DEL
npWebCapture.xpi DEL WebCapture.cab DEL WebCapture.crx
```

Updating Web Apps to WingScan post-10.5.2

In order to use WingScan (any version) you need to update the application's WebCapture Resources. These files can be anywhere in the app, but by convention, they would be together in a folder named WebCapture or WebCaptureResources

Wherever they are, to upgrade to 10.5.2, do the following... first find and remove all of the following files from your application's WebCapture Resources directory:

```
talaCaptureService.js atalaEzTwain.js atalaLeader.js atalaUpload.js atalaWebCapture.js
```

```
atalaWebCapture.min.js npWebCapture.xpi WebCapture.cab WebCapture.crx
```

Next, from the following location:

C:\Program Files (x86)\Atalasoft\DotImage 10.7\bin\2.0\x86\WebResources\WebCapture\

Copy these files into your application's WebCapture Resources directory:

```
talaWebCapture.js jquery-1.7.1.min.js Kofax.WebCapture.Installer.msi
```

Now, you can stop here and rebuild your app and it would be set to use the new WingScan. However, since the old WingScan was Plugin-based and the new one is using a local service, users who need to scan will need to actually install the service. - see the next section for that

Client-side Installation

The old plugins are no longer needed, but the client will need to install our new WebCapture service on their PC in order for the new WingScan to work.

All they need to do is to download and run

Kofax.WebCapture.Installer.msi

Note: it's now in your WebCapture Resources directory, so you can give them a direct link to download the installer.

If you had code in your app that was handling errors through the onScanError configuration, then you will want to ensure you update the handlers for the following msg items:

- Atalasoft.Controls.Capture.Errors.activeX
- Atalasoft.Controls.Capture.Errors.noPlugin
- Atalasoft.Controls.Capture.Errors.oldPlugin

The idea is that you would tell them that in order for scanning to function, they need to install the WebCapture service, and then give them a link to the

Kofax.WebCapture.Installer.msi

For more information on the installer, deploying the web capture service and troubleshooting, please see:

INFO: WebCapture Service Technical Overview

Breaking Changes in the JavaScript API

CORS implementation in Internet Explorer 8 / 9 does not support synchronous methods. To improve performance and to maintain support for IE8 and IE9, we provided new asynchronous overloads for all synchronous functions. The following table shows the new overload for all synchronous functions.

Synchronous function	New asynchronous overload
Image.saveEncryptedLocal(format [,opts])	Image.saveEncryptedLocal(format [
Image.asBase64String(fmt [,opts])	Image.asBase64String(fmt [,opts], c
LocalFile.asBase64String(fid [,fmt [,opts]])	LocalFile.asBase64String(fid [,fmt [,
LocalFile.remove(fid)	LocalFile.remove(fid, callback)
LocalFile.list()	LocalFile.list(callback)
LocalFile.removeAll()	LocalFile.removeAll(callback)
LocalFile.globalPurgeByAge(hrs)	LocalFile.globalPurgeByAge(hrs, ca
LocalFile.fromBase64String(key)	LocalFile.fromBase64String(key, ca
LocalFile.setEncryptionKey(str)	LocalFile.setEncryptionKey(str, callb

The new overloads simply take an additional parameter which is a callback function which will be invoked when the call completes.

Synchronous methods will be preserved in public JavaScript API for backward compatibility purposes and will work correctly in browsers that support synchronous CORS requests (Chrome, Firefox, Internet Explorer 10 and newer).

Synchronous methods will be marked as deprecated – they will write warning messages to the browsers console at runtime. Instead, new asynchronous versions of these APIs should be used. They will have the same signature as before with additional callback parameter that will be called on operation completion. Asynchronous APIs will work in all supported browsers.

Note: This is a breaking change for IE 8 and 9, the new overloads must be used. For all other supported browsers, we recommend the change is made to improve performance.

Original Article

Q10408 - HOWTO: Migrate a WingScan based app to 10.5.2 (and newer - includes 10.6, 10.7, 11.0 and 11.1)

Last Update

2026-10-12 - TD - (Added notice about applicability and fixed formatting issues)