Sage 300 Web Screens SDK

Upgrade Guide for 2021.2 to 2022.0

August 2021

The MIT License (MIT)

Copyright © 1994-2021 The Sage Group plc or its licensors. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Contents

[1. Overview 4](#_Toc80621797)

[1.1 Required Version of Sage 300 5](#_Toc80621798)

[2. Installing the Sage 300 Upgrade Wizard 6](#_Toc80621799)

[3. Accessing the Sage 300 2022.0 Upgrade Wizard 7](#_Toc80621800)

[4. Using the Sage 300 Upgrade Wizard 8](#_Toc80621801)

[4.1 Step 1 – Synchronize Kendo Files 9](#_Toc80621802)

[4.2 Step 2 – Synchronize Web Files 10](#_Toc80621803)

[4.3 Step 3 – Synchronize Accpac Libraries 11](#_Toc80621804)

[4.4 Step 4 – Remove previous version of jQuery libraries 12](#_Toc80621805)

[4.5 Step 5 – Namespace and Web Project Updates 13](#_Toc80621806)

[4.6 Step 6 – Finder Alterations 14](#_Toc80621807)

[4.7 Step 7 – Javascript Minification Updates 15](#_Toc80621808)

[4.8 Step 8 – Confirmation 16](#_Toc80621809)

[4.9 Step 9 – Recompile 17](#_Toc80621810)

[5. Upgrade Log 18](#_Toc80621811)

[6. Finder Alterations 19](#_Toc80621812)

[6.1 Overview 19](#_Toc80621813)

[6.2 Finder Architecture 19](#_Toc80621814)

[6.3 Screen Implementations 19](#_Toc80621815)

[6.3.1 Screen Implementation Steps (summary) 20](#_Toc80621816)

[6.4 Removal of Old Finders 21](#_Toc80621817)

[6.4.1 Finder Controllers 21](#_Toc80621818)

[6.4.2 Finder Registrations 21](#_Toc80621819)

[6.5 Web SDK Samples 21](#_Toc80621820)

[7. Javascript Minification Updates 22](#_Toc80621821)

[8. Compilation Troubleshooting 23](#_Toc80621822)

1. Overview

This document is intended to serve as a guide for illustrating how to use the Sage 300 Upgrade Wizard to upgrade Visual Studio solutions and projects that are compatible with the Web Screens SDK 2021.2 to Web Screens SDK 2022.0.

The wizard will upgrade the solution and projects in following steps:

* Synchronize Kendo Files
* Synchronize Web Files
* Synchronize Accpac Libraries
* Remove previous version of jQuery libraries
* Namespace and Web Project Updates
* Finder Alterations
* Javascript Minification Updates
  1. Required Version of Sage 300

To get started, install Sage 300 2022.0 with the Web Screens option selected.

1. Installing the Sage 300 Upgrade Wizard

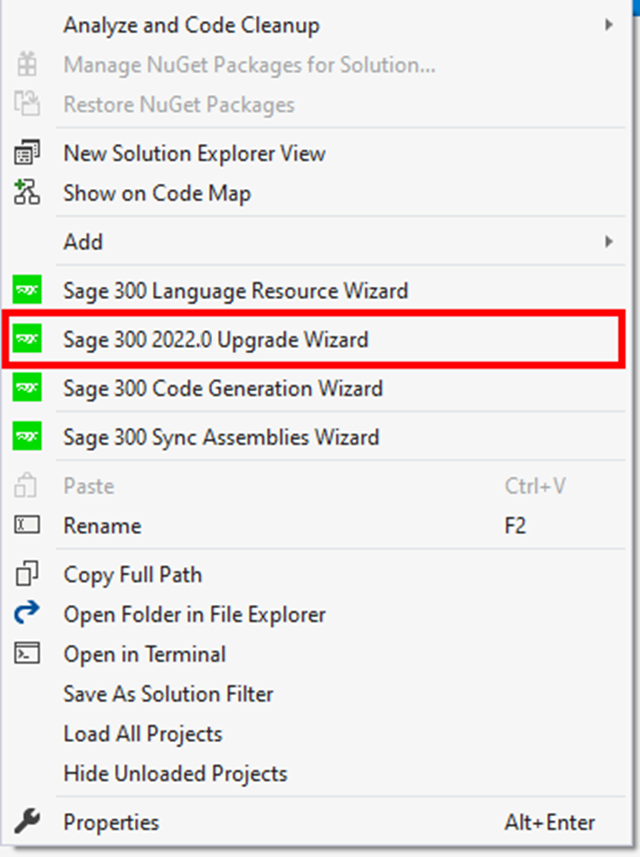
The Upgrade Wizard is a Visual Studio Plugin and is compatible with Visual Studio 2019.

**Note:** If the previous Upgrade Wizard is installed, it must be uninstalled first (In Visual Studio - Tools, Components and Extensions, Uninstall option once the Upgrade Wizard is selected).

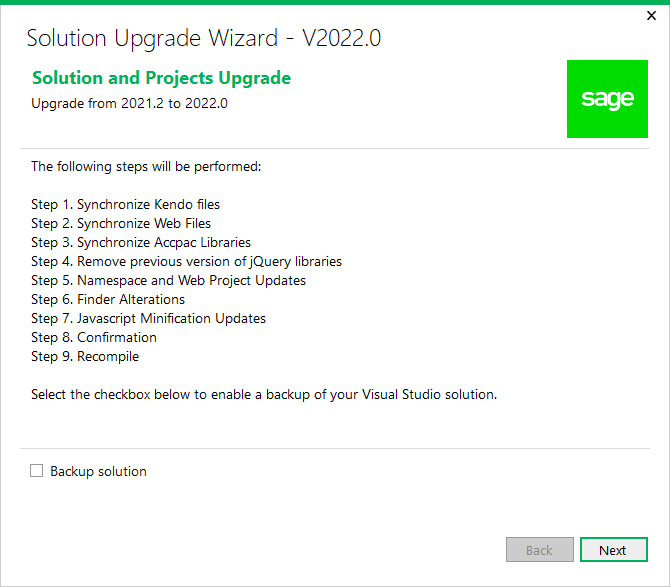
To install the wizard:

1. Run Sage300UpgradeWizardPackage.vsix.
2. Select the Install the Wizard Package option.
3. Accessing the Sage 300 2022.0 Upgrade Wizard

The **Sage 300 2022.0 Upgrade Wizard** is a Visual Studio Plugin. To open it, with the partner solution loaded in Visual Studio, right-click the solution, and on the context menu, click Sage 300 2022.0 Upgrade Wizard.



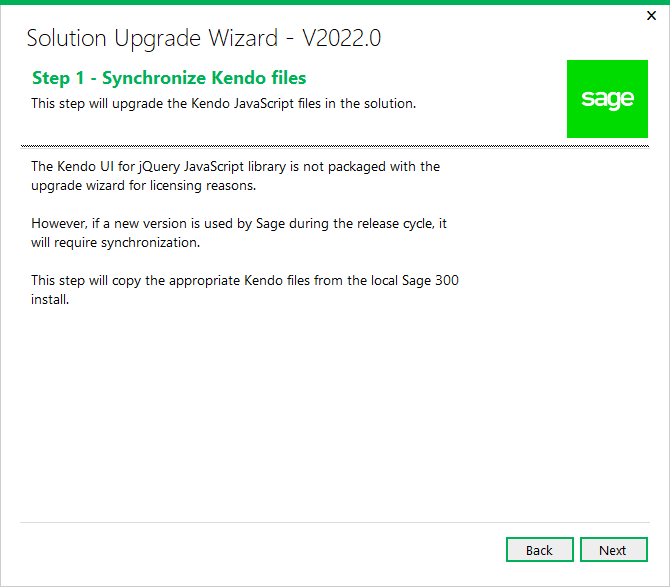
1. Using the Sage 300 Upgrade Wizard

­

**Important:** Please backup the solution and projects before proceeding with the upgrade. If you would like the Upgrade Wizard to make a backup of your solution, click the ‘Backup solution’ checkbox.

Click **Next** to continue to the next step.

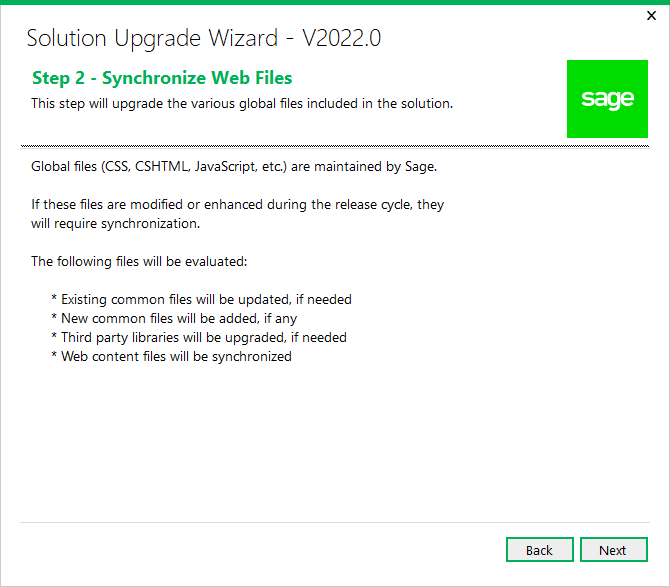
* 1. Step 1 – Synchronize Kendo Files



A copy of the Kendo UI for jQuery JavaScript library will be copied into the solution from the local Sage 300 installation.

Click Next to proceed or Back to go back to the previous step.

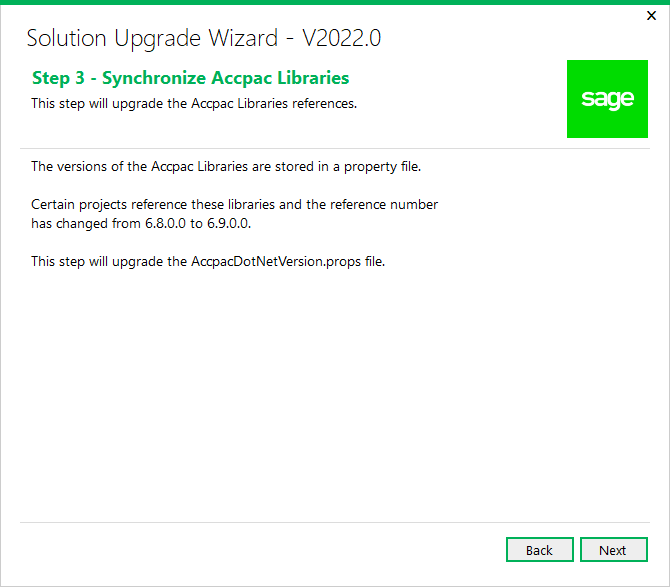
* 1. Step 2 – Synchronize Web Files



Contents in the Web Projects Areas\Shared, Areas\Core, Views, Scripts, Content, and Assets folders are upgraded in this step.

Click Next to proceed or Back to go back to the previous step.

* 1. Step 3 – Synchronize Accpac Libraries



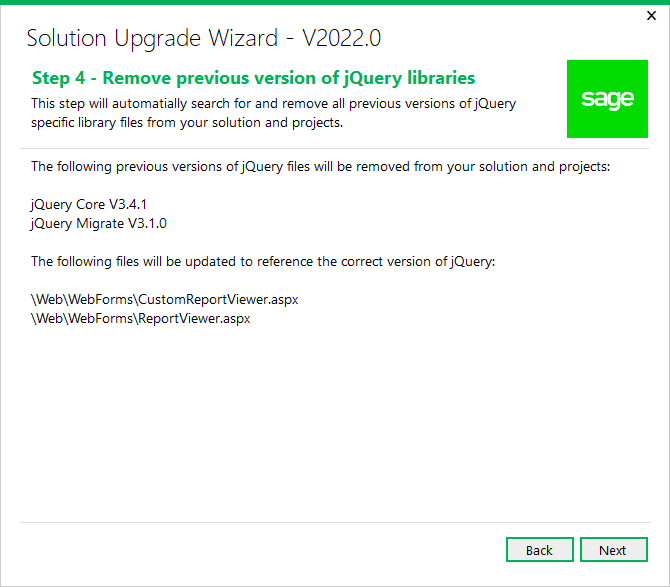
The upgrade wizard will search for copies of the **AccpacDotNetVersion.props** file in the following locations:

* + Root of the Solution folder
    - If found, it will stop looking. If not found, it will then search in the root of each project folder within the solution.

If the **AccpacDotNetVersion.props** file is found in the root of the solution folder, it will be upgraded as normal. If the **AccpacDotNetVersion.props** file is not found in the root of the solution folder but is found in one or more project folders, the wizard will update the project file (.csproj) in each of these folders with the correct path to the **AccpacDotNetVersion.props** file located in the solution folder and then remove the copy of the **AccpacDotNetVersion.props** file in the project folder. When this process has completed, the solution should contain only a single **AccpacDotNetVersion.props** file located in the root of the solution folder.

Click Next to proceed or Back to go back to the previous step.

* 1. Step 4 – Remove previous version of jQuery libraries



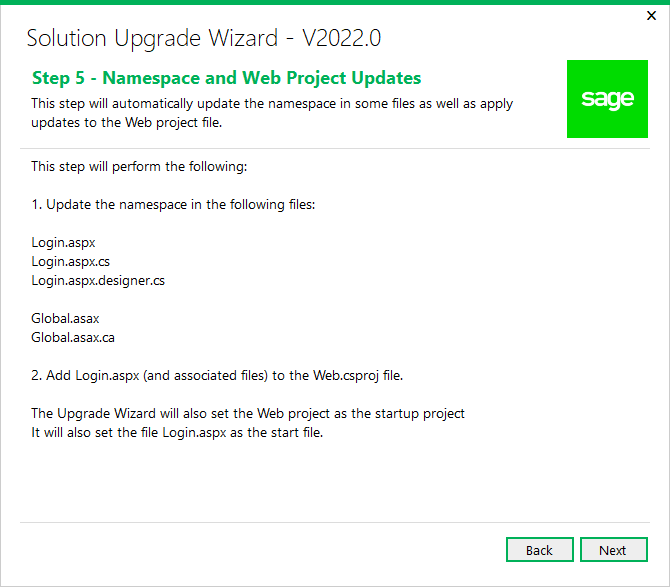
The upgrade wizard will automatically remove the following:

* + jQuery Core V3.4.1
  + jQuery Migrate V3.1.0

The upgrade wizard will also update some of your solution files to reference the correct version of jQuery. The 2022.0 release uses jQuery V3.6.0.

Click Next to proceed or Back to go back to the previous step.

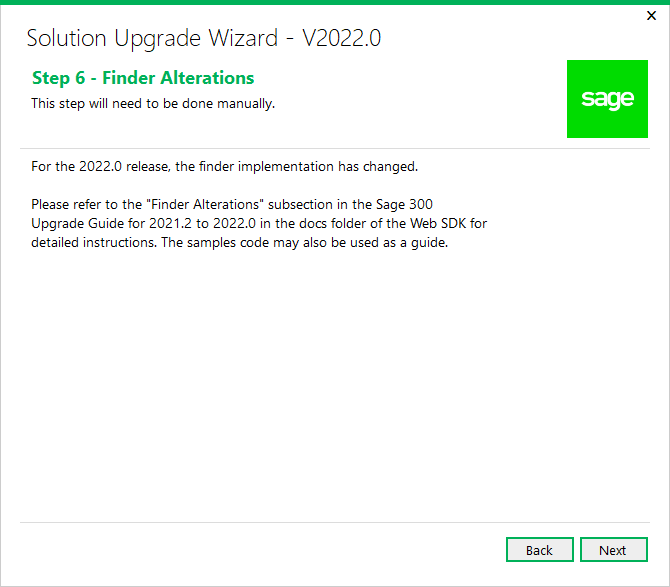
* 1. Step 5 – Namespace and Web Project Updates



The upgrade wizard will determine the root Web namespace and update a select list of files with this namespace. Once that has been done, the upgrade wizard will add the file Login.aspx and associated related files to the Web project.

Click Next to proceed or Back to go back to the previous step.

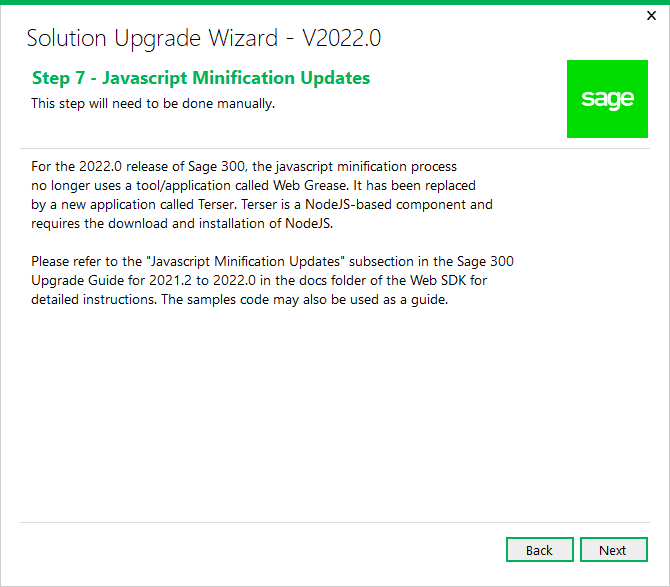
* 1. Step 6 – Finder Alterations



This is simply an informational step. See **Section 6 – Finder Alterations** for full details on the manual modifications required.

Click Next to proceed or Back to go back to the previous step.

* 1. Step 7 – Javascript Minification Updates

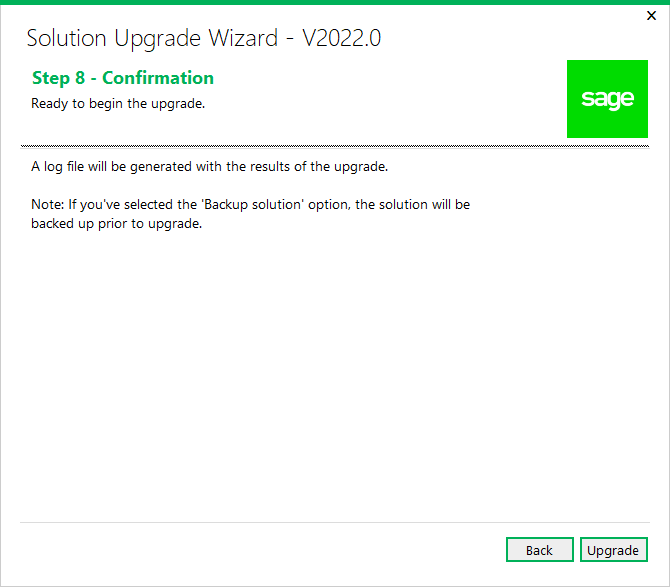


This is simply an informational step. See **Section 7 – Javascript Minification Updates** for full details on the manual modifications required.

Click Next to proceed or Back to go back to the previous step.

ddddd

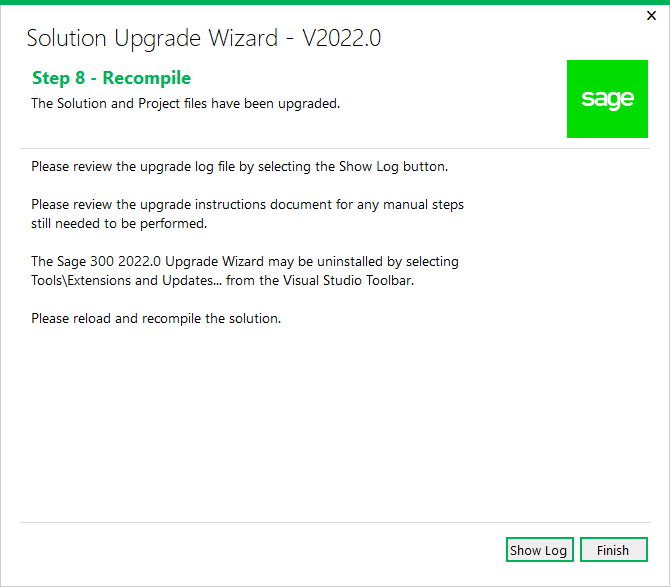
* 1. Step 8 – Confirmation



The solution and projects are ready to be upgraded.

Click Upgrade to upgrade the solution and projects or Back to go back to the previous step.

* 1. Step 9 – Recompile



Click **Show Log** to display log file. The log file displays what files have been upgraded or modified. At this point, the upgrade is now complete.

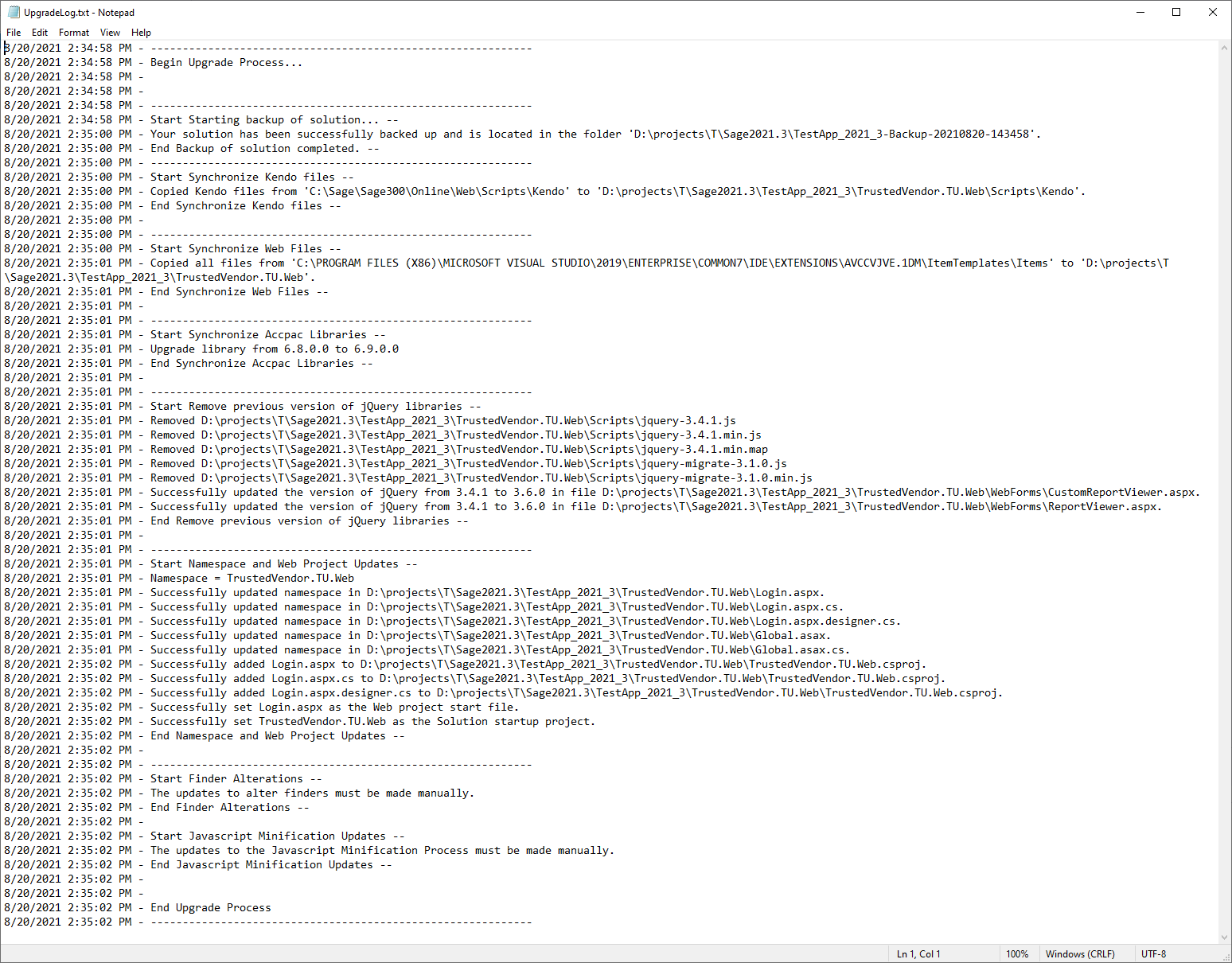
Click **Finish** to exit the wizard.

After the solution that supported Sage 300 2021.2 has been upgraded to support 2022.0, you need to reload and recompile the solution. For compilation troubleshooting, see the next section.

**Note:** Before running the new solution, clear the browser cache to replace cached JavaScript and CSS files.

1. Upgrade Log

Once the Upgrade Wizard has completed you have the option to ‘Show Log’ to view the upgrade log. The following is a screenshot of a typical upgrade log.



1. Finder Alterations
   1. Overview

When the web screens were originally developed, a web finder was implemented that was different from the desktop finder for various reasons. This “original” implementation had numerous issues; such as, it contained too much code, there was too much redundant and complex code, customization and dynamic creation were difficult or not possible.

A few releases back, the screens introduced a new and improved web finder that reduced the amount of code required to configure a finder, interacted directly with the Accpac view (for performance and simplicity), and even allowed for dynamic creation. While this “new” finder has co-existed for several releases with the old finder, the release of Sage 300 2022 introduced significant improvements to this web finder that were not made in the “original” web finder.

For consistency the “original” web finder framework has been removed from the application and all “original” web finders have been replaced with new web finders. This effort constituted two-phases for Sage 300 web screens in 2022:

* Replace the finder implementation per screen
* Deprecate and remove the old finder definitions and registrations

This section will document the steps required to replace a finder in a screen and for “original” partner created finders to be removed.

* 1. Finder Architecture

In the Web SDK in the docs/development folder, the **Sage300SDK\_FinderArchitecture** document shows the “original” interface, explains the “new” interface, finder configurations, and provides some examples of the new implementation. Please refer to this document.

* 1. Screen Implementations

As noted in section 6.2, please refer to the finder document for the finder interface and examples.

If a Sage 300 “original” finder was being invoked (i.e., sg.setFinder…) in a partner screen, this Sage 300 finder no longer exists as it has been removed along with the finder framework that supported it. The “new” definition resides in the ViewFinderProperties.js file. Here are a couple examples of new finder implementations that invoke a finder based upon a definition in the ViewFinderProperties.js file:

1. sg.viewFinderHelper.setViewFinderEx("btnFindVendorNumber", "Data\_VendorNumber",

           sg.viewFinderProperties.AP.APVendor, onFinderSuccess.VendorNumberFinder, $.noop);

1. const taxClassesFinderInfo = () => {

           const property = sg.utls.deepCopy(sg.viewFinderProperties.TX.TaxClasses);

            const selectedRowData = sg.utls.kndoUI.getSelectedRowData(vendorGridUtility.fetchTaxAuthorityGrid());

            const authority = selectedRowData.TaxAuthority;

            const classType = 2;

            const classAxis = 1;

            property.initKeyValues = [authority, classType, classAxis, selectedRowData.TaxClass];

            property.filter = $.validator.format(property["filterTemplate"], authority, classType, classAxis);

            return property;

        };

        sg.viewFinderHelper.setViewFinder("btntaxclassfield", onFinderSuccess.OnTaxClassSelection,

            taxClassesFinderInfo, onFinderCancel.onTaxClassCancel);

You will need to revise your *initFinder*(), *setFilter*(), and call back routines in the screen’s …Behvaiour.js file to not only invoke the new finder but also to support the new behavior since the new finder’s are bound to the finder widget and interact directly with the widget.

Please also refer to the samples in the Web SDK as examples as their finders have been replaced with new finders.

If you are a partner or ISV that has created a module/application and your screens are invoking your finders, the new configurations must be created first in your version of ViewFinderProperties.js. A new Wizard has been created for 2022 called the Finder Definition Generator (available in Visual Studio as a plugin (part of the *Sage300UIWizardPackage.vsix*)) that allows finder definitions to be created. Please see the Finder Definition Generator Document in the Web SDK in the docs/wizards folder.

When creating the partner finder definition, please reference your finder defined in the Controller/Finder folder of your web project for the configured columns. The finder controllers will be removed in the next section.

* + 1. Screen Implementation Steps (summary)
* Replace any Sage 300 Finder (AR, AP, GL, etc.) invocation with code that invokes the new Sage 300 Finder based upon sg.viewFinderProperties.{*module*}.{*name*} configuration
  + See samples in the Web SDK for examples
* For partner created finders:
  + Invoke the Finder Definition Generator in Visual Studio to create a finder definition
  + This utility will add the definition to your version of ViewFinderProperties.js
  + Your version of ViewFinderProperties.js will be required to be added to the JavaScript bundle for your screen in the *BundleRegistration*.cs file of your web project
  + Replace any partner finder invocation with code that invokes the new partner finder based upon the newly created configuration in your version of ViewFinderProperties.js
  1. Removal of Old Finders

Sage 300 release 2022 has removed all the “original” finder framework, all the Sage 300 module finders, and all the finder bootstrapper entries.

The Upgrade Wizard for 2022 has modified the partner solution to remove the “original” finder framework components from the partner solution/web project.

The section will explain the files and bootstrapper entries that now must be deleted for the application to compile and run successfully.

* + 1. Finder Controllers

In your web project, you will see the “Finder” folder under your “Areas/{*module*}/Controllers folder. This folder contains the files for the “original” finder definitions (i.e., *FinderVendorControllerInternal*,cs).

These finders are now configured/defined in the ViewFinderProperties.js file equivalent and therefore these files are now obsolete and must be deleted.

* + 1. Finder Registrations

In your web project, you will see the “{*module*}WebBootstrapper.cs” file. This file contains the Unity registrations for the finder controller’s that were just deleted in step 6.4.1.

These registrations must be deleted as they are no longer required.

Failure to delete these registrations will cause a failure when running the partner solution.

* 1. Web SDK Samples

Although it was mentioned several times in the proceeding sections, the Web SDK Samples have been updated to invoke the “new” web finders.

These samples provide numerous examples of invoking finders and can be used as a roadmap to assist with the manual replacement of “original” finders with “new” finders.

1. Javascript Minification Updates

For the 2022.0 release of Sage 300, the javascript minification process no longer uses a tool/application called Web Grease. It has been replaced by a new application called [Terser](https://terser.org/). Terser is a NodeJS-based component and requires the download and installation of NodeJS.

**Instructions for downloading/installing NodeJS**

* + Download the NodeJS installer from <https://nodejs.org/en/download/> and select your target operating system.
  + Follow the instructions in the installer to complete the setup process.

1. Compilation Troubleshooting

**Parser Error when loading Web project in browser**

* Clean solution
* In Web project folder:
  + delete bin and obj folders