embarcadero®

Product Documentation

Embarcadero® AppWave™ Studio 2.0
User Guide

Embarcadero Technologies, Inc. is a leading provider of award-winning tools for application developers and database professionals so they can design systems right, build them faster and run them better, regardless of their platform or programming language. Ninety of the Fortune 100 and an active community of more than three million users worldwide rely on Embarcadero products to increase productivity, reduce costs, simplify change management and compliance and accelerate innovation. The company's flagship tools include: Embarcadero® Change Manager™, CodeGear™ RAD Studio, DBArtisan®, Delphi®, ER/Studio®, JBuilder® and Rapid SQL®. Founded in 1993, Embarcadero is headquartered in San Francisco, with offices located around the world. Embarcadero is online at www.embarcadero.com.

June 15, 2011

Contents

WELCOME TO EMBARCADERO APPWAVE STUDIO

The AppWave Studio is an application within the AppWave Browser where you can master an application for AppWave as an app.

Apps are the applications that are mastered into a single file that runs without installation. The app runs on the desktop, locally and under 'full power' but does not change the operating system or registry. Apps run within their own space, with no shared DLLs or other system-wide settings that could conflict with other apps. Application data, including settings and licenses are stored locally on the hard drive separate from the executable.

Once your app is broadcast, an app build is placed in a working folder. This build is a copy of the normal install build which is compressed into one single standalone executable file that runs instantly on any desktop. This process takes the entire structure of an installed app with all the settings (registry, and default settings of the application) and combines them into the final standalone build. This "build" can be executed from a network without being installed on that computer.

The advantage of using the App technology is that the app is broadcast on the system and then when it is removed, the system will be in a state as if it was never installed. Any file or registry keys that are created by the creation of the app are deleted from the file system on removal. If the app makes modification to a file on the system, the original state is restored.

> **NOTE:** For additional information and support on Embarcadero's products, please refer to the Embarcadero web site at http://www.embarcadero.com.

This chapter covers the following topics:

- AppWave Studio Users
- Technical Requirements
- Mastering Apps
- Benefits of Using Apps
- Private vs. Public AppWave

APPWAVE STUDIO USERS

AppWave Studio has two different types of users: Administrator and User:

Administrator: If you have Administrative privileges you have access to all the AppWave functionality. When you first log onto the AppWave Browser an Admin tab appears in the header.

The Administrator:

- Has access to all the administrative functions in AppWave. The Administrator can access the functionality through a web browser or through the Admin tab in the AppWave Browser.

- Has the option to give users access to the AppWave Studio so they can master apps. For more information, please refer to the **Group Management** topic in the *AppWave Administrator Guide.*

- Can collect and analyze product usage on a per user, group, and product basis.

- Can broadcast multiple product configurations and assign them to individuals and groups.

- Can retrieve updates from the public server or other privately deployed servers.

- Manages license inventory, assignments, and connected users. Also ensures the correct information is available for License Automation. For more information, please refer to the **Manage Licenses** topic in the *AppWave Administrator Guide.*

Users: When a user opens the AppWave Browser they have access to the App tab where they can view and run apps locally on their own desktop without installation.

The user:

- Is added to the **All Users** group.

- Has access to supported apps as easy-to-use alternatives to installing and configuring apps. These users can create an app from a template in the AppWave Browser. Once the app is mastered, it can then be streamed from the AppWave and then run locally on the desktop without installation.

- If granted authoring permission from the Administrator, has access to the AppWave Studio where they can create Apps two different ways: (1) from an executable that automatically captures file and registry changes; or (2) by building an app manually where they insert required information.
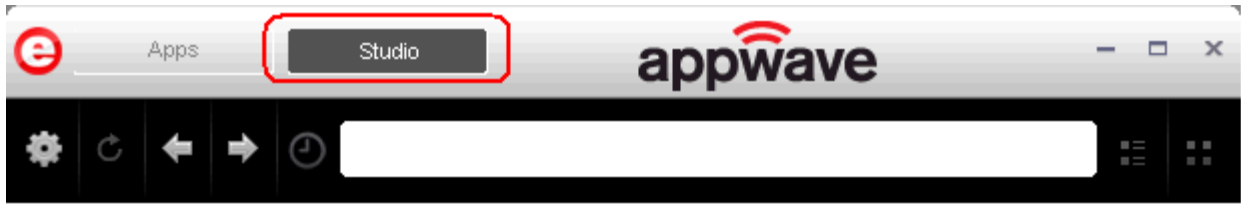
TECHNICAL REQUIREMENTS

AppWave Studio is an application that resides in the AppWave Browser. The technical requirements necessary to run AppWave Studio are the same as is required for the AppWave Browser.

For more information on the AppWave Browser requirements, see *Technical Requirements* in the *AppWave Browser User Guide*.

AppWave Browser supports 32-bit applications on 32-bit and 64-bit operating systems. 64-bit applications are not supported.

> **NOTE:** Mastering Apps in Studio is currently supported only on 32-bit operating systems.

AppWave Browser is 41 MB in size and occupies a minimum of 70 MB in memory initially. More memory will be consumed based on usage.

Default cache size for AppWave Browser is 2148 MB on Windows XP and 6144 MB on Windows 7 64-bit. The size of the cache varies on the operating system used and the amount of memory available.

Products available in the AppWave Browser have the same approximate memory and disk size requirements as they do independent of All-Access. For the requirements of individual Embarcadero products, consult the product documentation at http://docs.embarcadero.com.

For more information on the AppWave requirements, see: *Installation Information* in the *AppWave Administrators Guide.*

MASTERING APPS

An app is an application that has been mastered with AppWave and then made available in the AppWave Browser.

> **NOTE:** Some applications are not suitable for use with AppWave Studio and AppWave, for example an applications that must run as a Windows service. Since by definition they start automatically as a service when you boot up Windows, they are not able to run on a different machine.

Also if an application is not suitable if it is configured to run against a database or other service that is installed on the same machine, for example, an application that locally installs a database and the database connection is hard coded. Applications such as this may have the option to be installed as a network client and can be configured to access a database on the network; that configuration is more likely to work as a Studio App.

CAUTION: During mastering, extra services related to the app you are mastering may be launched. At the end of the mastering process, Studio will attempt to shutdown any services it detects, but in some cases Studio cannot detect those services and you must shut them down manually before broadcasting the App.

There are several different apprFLoaches to creating an app depending on the source of the executable/installer used:

- **Create an App with One Click.** All the mastering support has been done. This process uses a template that has been created with the information needed to complete the App. This information includes descriptions, display name, product name, etc. All you need to do is supply an application's executable. This creation process starts in the Apps portion of the AppWave Browser. You then complete the creation process in the Studio portion.

  User: An Enterprise Desktop Administrator - Only basic programming skills are necessary. The process is streamlined with many of the necessary decisions made for the user.

- **Create AppWave Supported or Custom App from Install.** With this option, you create, edit, and broadcast an app from Install using AppWave Studio. The steps to be completed depend on the application you have chosen to install; some steps may be automatically completed for you.

  User: An Independent Software Vendor (ISV) user - The user mastering the application must have the knowledge to make key decisions during the creation process. This includes knowledge about the results of decisions made and when and why to use the default recommendations.

- **Create AppWave Supported or Custom App from Scratch.** An application that does not have the mastering support. This means that you would need to manually insert Product Name, Display Name, Version, Long Description, etc.

  User: The user needs registry knowledge. This user also needs an understanding of the installation process including files (location and properties) and environment variables.

- **Create an App with Multiple Versions.** When you master an app using an AppWave Template, that template may contain multiple versions of the application. If that is the case, you have the option select the version, and to either open or record that version.

  User: The user needs registry knowledge. This user also needs an understanding of the installation process including files (location and properties) and environment variables.

For more information, see Using AppWave Studio.

BENEFITS OF USING APPS

- The Apps are streamed to your computer and not installed so the app doesn't take up space on your computer's hard drive.

- Any files or registry keys that are created during the mastering process, as well as any modifications made to files on the system, are deleted when the system is removed. This means that the original state is restored.

- There is no application conflict – You can determine whether applications are fully integrated with your environment, fully isolated, or any combination between the two. This prevents conflicts between applications. You can run two separate versions of the same application without conflict.

- Application data such as configuration and license information is stored on your hard disk and is separate from the application's executable.

- You can gain access your app set from any computer.

- You always have access to the latest version of an app – You don't have to keep track of install patches, service packs, upgrades, etc.

For more information, see: *Understanding App Products* in the *AppWave Browser Apps User Guide*.

PRIVATE VS. PUBLIC APPWAVE

The Private AppWave determines what content can be received from the Public AppWave. Access to Apps are controlled by an AppWave administrator or through license enforcement.

If an app has not been streamed to the private AppWave from the public it is still possible for the end user to see and access the app. If an app that has not been streamed from public is launched by the end user, the app is streamed to the private AppWave and then to the end user. An Administrator may choose to stream the app to the private AppWave to optimize the availability.

The location of the app, whether public, private, or on the local machine, concerns the end user prior to launch because it tells the user of what type of delay may be expected. The status of the program loading can be defined by:

- **Unavailable** – in the case where a private server is used, this means the app has not been streamed to the server and the private server cannot currently access the public server.

- **Internet Availability** – This is when the private server does not have the app and it needs to be streamed down from the public server.

- **Remote Availability** – The private server has the app streamed.

- **Available** – The app is available on the local machine.

Private AppWave

A key scenario for AppWave is to help companies distribute apps (including open source, third party, and internal apps) through their user base. Rather than waiting for a new OS image with the company-picked set of tools, users can run without installing any company-approved app in minutes. At the same time, the company can manage the number of licenses currently in use and make sure they comply with the vendor rules.

You can install and manages your own server. The difference is allowing the addition of custom and third-party apps with the Embarcadero ones.

Public AppWave

The Public AppWave contains all of the Embarcadero mastered apps and third-party apps that can be distributed with valid licenses.

This is a public apps-management server managed by Embarcadero. An AppWave Host Administrator determines what is made available on the public server.

The public AppWave Host has the following major systems in its architecture:

- User Management

- License Management

- Product Catalog Management

USING APPWAVE STUDIO

The AppWave Studio allows you to create app files. Once you have created an App, it can be streamed from the AppWave and then run locally on your desktop without installation. This means that apps to be run on locked-down desktops. There are no conflicts with other apps and no writing to the system areas of their registry or modifying shared dlls.

The following topics are covered:

- Studio Access
- Best Mastering Practices
- Create an App with One Click
- Create AppWave Supported or Custom App from Install
- Create AppWave Supported or Custom App from Scratch
- Create an App with Multiple Versions
- Open an Existing Project

STUDIO ACCESS

When you open the AppWave Browser and only the Apps tab is visible, this indicates that you do not have permission to master apps using Studio.

To gain access to the Studio, the Administrator must grant you permission as an Author. Once this is done, the Studio button appears in the AppWave Browser.

For more information on how to set up this permission, see the Group Management.

Once you have been granted access, click **Studio.** This is where you can master apps and then use them in your organization.
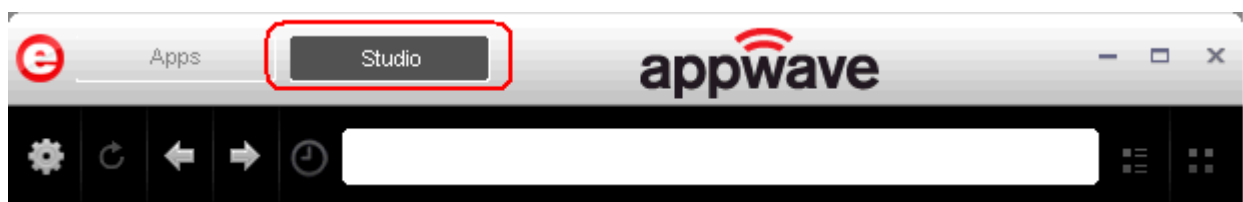
BEST MASTERING PRACTICES

- When mastering an app, the recommended location for installing the app is C:\application_name_version. Avoid installing in the default Program Files location as problems might occur when launching the app on a 64-bit system.

- Before mastering an app in AppWave Browser you must ensure that no other apps are currently running. If there are any apps running you will get a warning dialog when you click Master.

- Folders that point to a network location such as My Documents should be avoided. It is recommended that you set the network folder to a local folder.

- During mastering, extra services related to the app you are mastering may be launched. At the end of the mastering process, Studio attempts to shutdown any services it detects. But in some cases Studio cannot detect those services and you must shut them down manually before broadcasting the App.

- When trying to master an app using a template without installing, the app needs to already be on the system and installed with all the default options. If the app is installed in the Program Files folder and has this path in the configuration files for its usage or as part of other required components such as Java, there may be problems running the app on 64-bits systems.

- In order to master an app in Windows 7 or Vista, you must launch the AppWave Browser as an Administrator. To do this close the AppWave Browser, and open it again by right-clicking the AppWave Browser shortcut, and selecting "Run as Administrator".

- When mastering an app on the Windows 7 OS, create registry keys under HKEY_CLASSES_ROOT or in one of the subdirectories of HKEY_CURRENT_USER, HKEY_LOCAL_MACHINE, and HKEY_USERS.

  If the properties for the key you created are set to "All Applications" and "Permanently Replace Host Version" an error message (APM_E_APP_CONFIG_INSTALL_REGISTRY) will be generated.

- It is important to note that some applications will not work as a streamed app because of their product architecture:

  - Applications that must run as a Windows service. Because they start automatically as a service when you start Windows, they cannot run on a different machine.

  - Applications that are configured to run against a database or other service that is installed on the same machine. An example would be an application that locally installs a database and the database connection is hard coded. If an application has the option of installing as a network client and can be configured to access a database on the network; that configuration is more likely to work as a Studio App.

CREATE AN APP WITH ONE CLICK

You can use the AppWave Browser to create an app with one click. This process uses an app, available on the Apps tab, that has an AppWave template. AppWave templates have been created by Embarcadero for Embarcadero and third-party apps and ship with AppWave. This template tells AppWave how to master an app and contains branding information and information on which filter to run on the installer for the app, so unnecessary files can be filtered.

These apps with an AppWave template are considered "premium apps" and contain licensing information allowing for license automation which allows for activation of third-party licenses. AppWave then manages those licenses in the same way that Embarcadero application licenses are managed.

Apps that can be mastered are indicated by a icon as well as a designation of "Master".
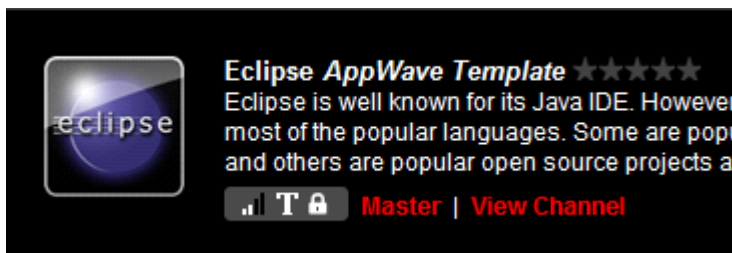
TIP: When you create an app with One Click you start from the Apps tab in the AppWave Browser and then complete it in AppWave Studio.

There are four major steps:

- Licensing
- Preparation
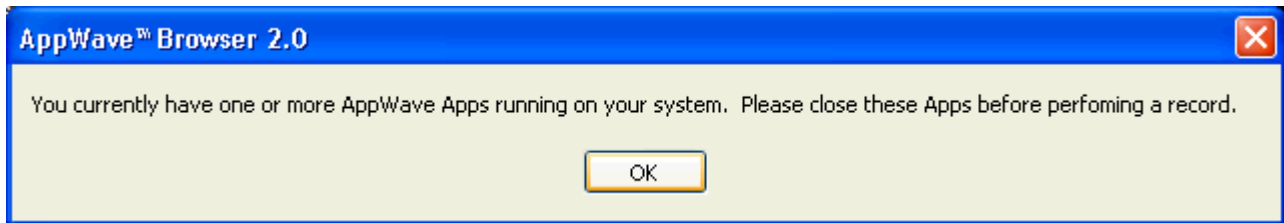- Setup
- Recording
- Running the App

LICENSING

The Administrator sets up the licensing for third-party apps through AppWave. Once the licenses are set up and activated, when the user masters the premium app the license automation function completes the licensing process.

For complete step-by-step instructions for this Administrative function, see: Licensing Third Party Applications.

PREPARATION

Before mastering an app in AppWave Browser you must ensure that no other apps are currently running. If there are any apps running you will get the following warning dialog when you click Master:

To close apps currently running:

> **NOTE:** You must stop each app separately.

1 In the AppWave Browser click **Activity** and all apps currently running are displayed.

2 To stop an app, right-click the app icon and select **Stop Application(s)**.

SETUP

1   In **Apps,** search (🔍) for "AppWave Template" and all apps with templates are listed. This template is indicated by an icon (**T**) located to the right of the app icon.
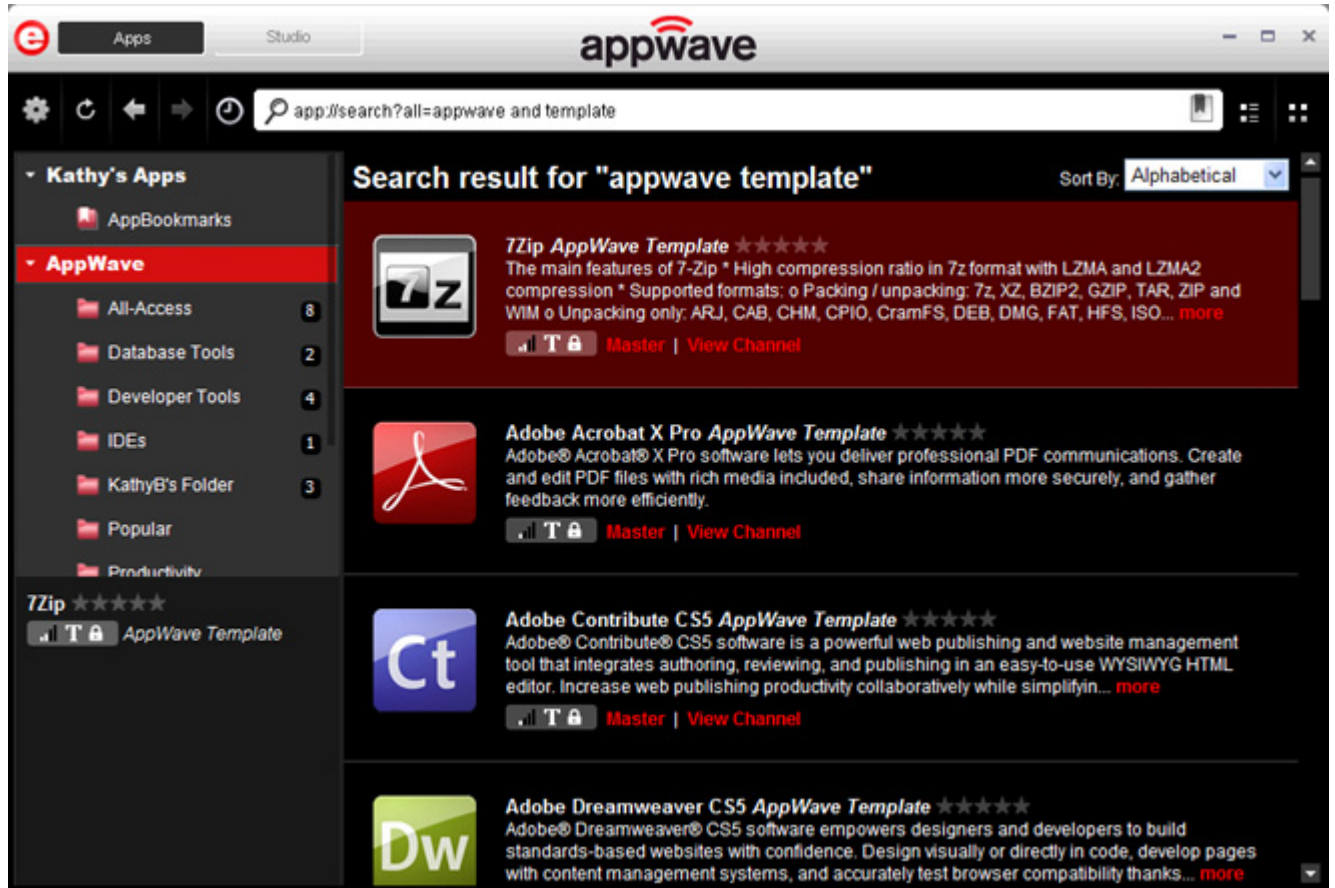
NOTE: If you are searching for a specific app or a type of app, you can use the Search field and all apps matching the search parameters appear in the field.

2   Scroll down and select the app that you want to master. In the following example we scrolled down to "FileZilla" which has an AppWave Template.
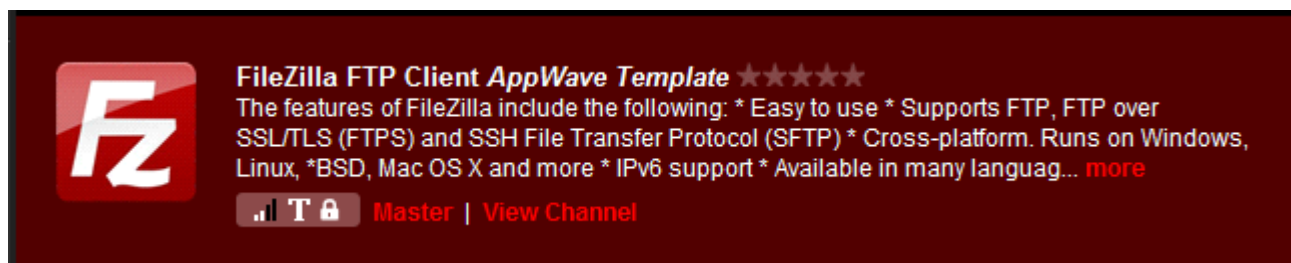
3   Click **Master** located to the right of the icon.

> **NOTE:**   A indicator window opens in the lower right-hand corner of your screen showing that the selected file has finished streaming. The file extension .aw indicates it is the AppWave Template file.

4   The **Record Install** dialog opens briefly followed by the **Record <application name>** dialog. Browse to the location of the executable. The web site information is automatically populated.

   NOTE: Click the web site URL if you have not downloaded the executable for the selected app you are going to master. This executable needs to be saved in the `C:\` directory.
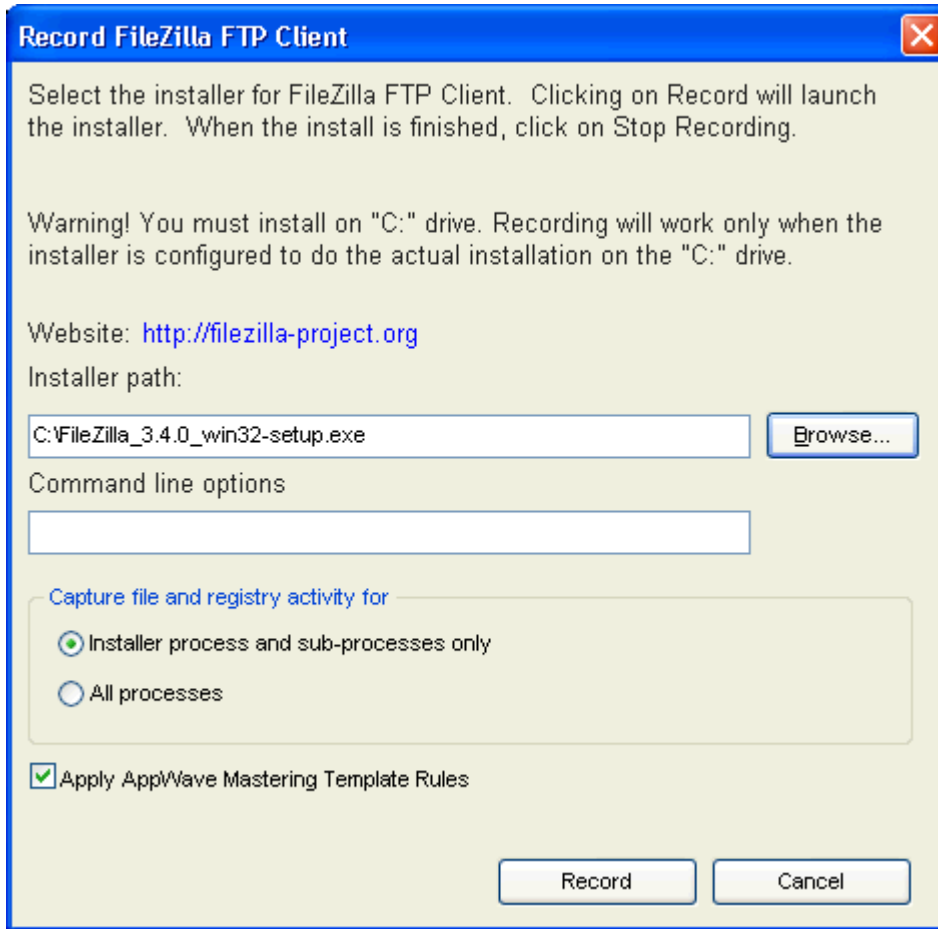
This dialog has the following fields:

Field	Description
Installer Path	Click Browse to find the executable for the app you want to master.
Command Line Options	The information entered here indicates how to launch the file and is the trigger to run a specific command.
	For information on commands you can use, see Using the Command Line.

Field	Description
Installer process and sub-processes only	This setting is selected by default. This option captures only the install processes.
All Processes	Captures all the processes associated with the app.
Apply AppWave Mastering Template Rules	This option is selected by default. As the process goes through the steps to create an App, tasks may be created indicating missing data. This option automatically competes any missing data.

RECORDING

1 Once you have completed the information click **Record** and the install process starts.

2 A Recording Installation dialog briefly appears.

3 A Recording dialog appears in the upper right corner of your screen and the AppWave Browser is minimized. The blinking recording status button indicates that recording is in progress.
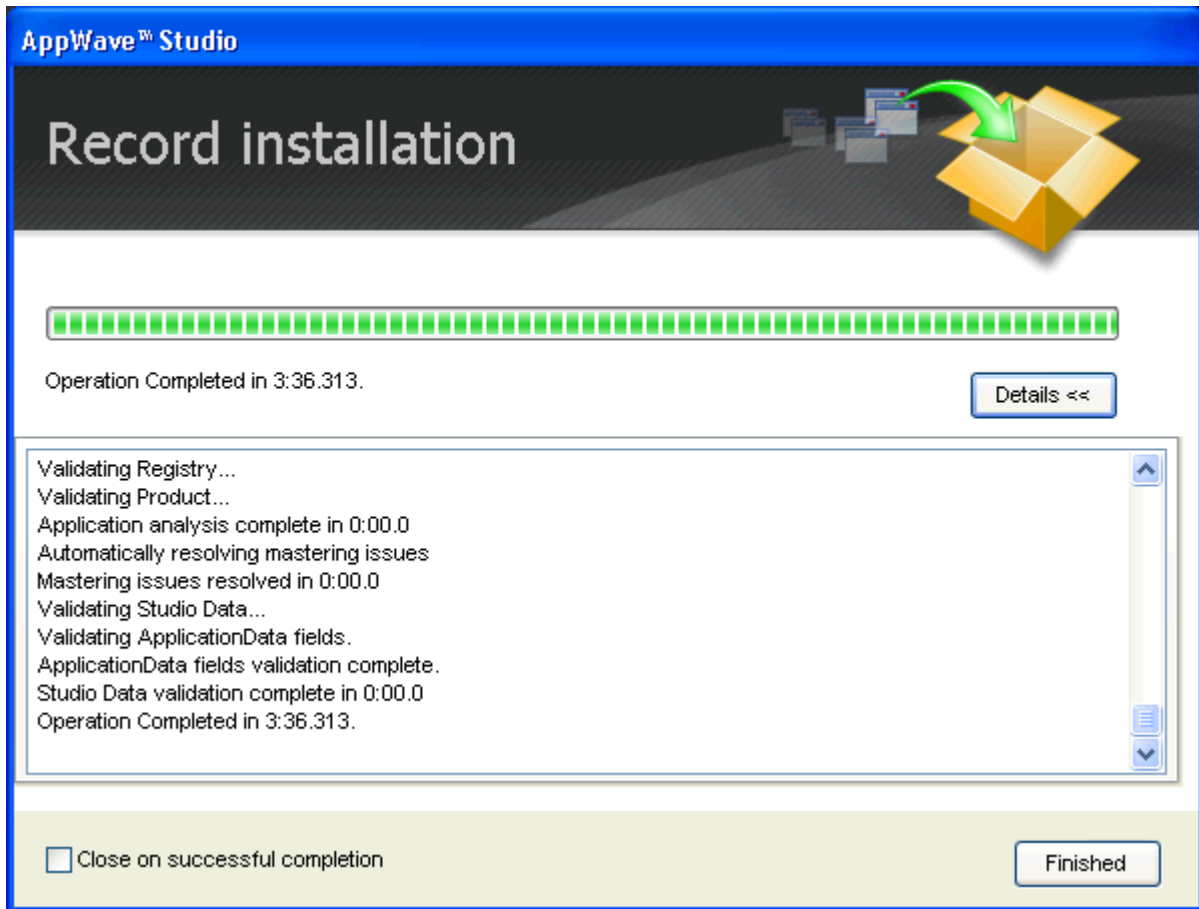

4 Follow the install wizard for your selected app.

5 The licensing of your app depends on what type of AppWave licensing you have purchased as well as the type of app you are creating. For a full explanation of how to license your app, please refer to: Managing Licensing.

6 On the final panel of the Setup wizard, you may have the option to launch the app. Make your selection and click **Finish** to close the wizard.

> **TIP:** You may not want to select the Launch option so you can investigate the registry keys that are created. If the roots of these keys are not in the app it is a good idea to include them so they will be managed.

7   Once the installation is complete, click **Finished** in the Recording dialog and the **Record Installation** dialog appears. This dialog identifies everything that needs to be installed. A progress bar shows each step of the process.

> NOTE: Click **Details** to see the details of the installation while it is broadcasting or after it completes. If you want to see it after the installation is complete deselect the "Close on successful completion" option and then click **Details**.

8   If this dialog box does not close automatically click **Finished** in the Record Installation dialog and one of several options are available depending on the template and executable in use.

> NOTE: At this point in the mastering process you are now working in the Studio portion of the AppWave Browser.

- The Launch page is where you supply the name of the launch executable. If this page appears you may need to supply information for the required fields such as the executable and the working folder.

OR

- The Brand page is where you provide branding information for the app. If this page appears you need to supply missing information such as Publisher, Display Name, or Product Name.

OR

- The Broadcast page is where you broadcast your mastered app to the AppWave. This page appears when the information is complete in the previous five steps and require no action on your part before broadcasting.

9   To complete this last step click **Broadcast**.
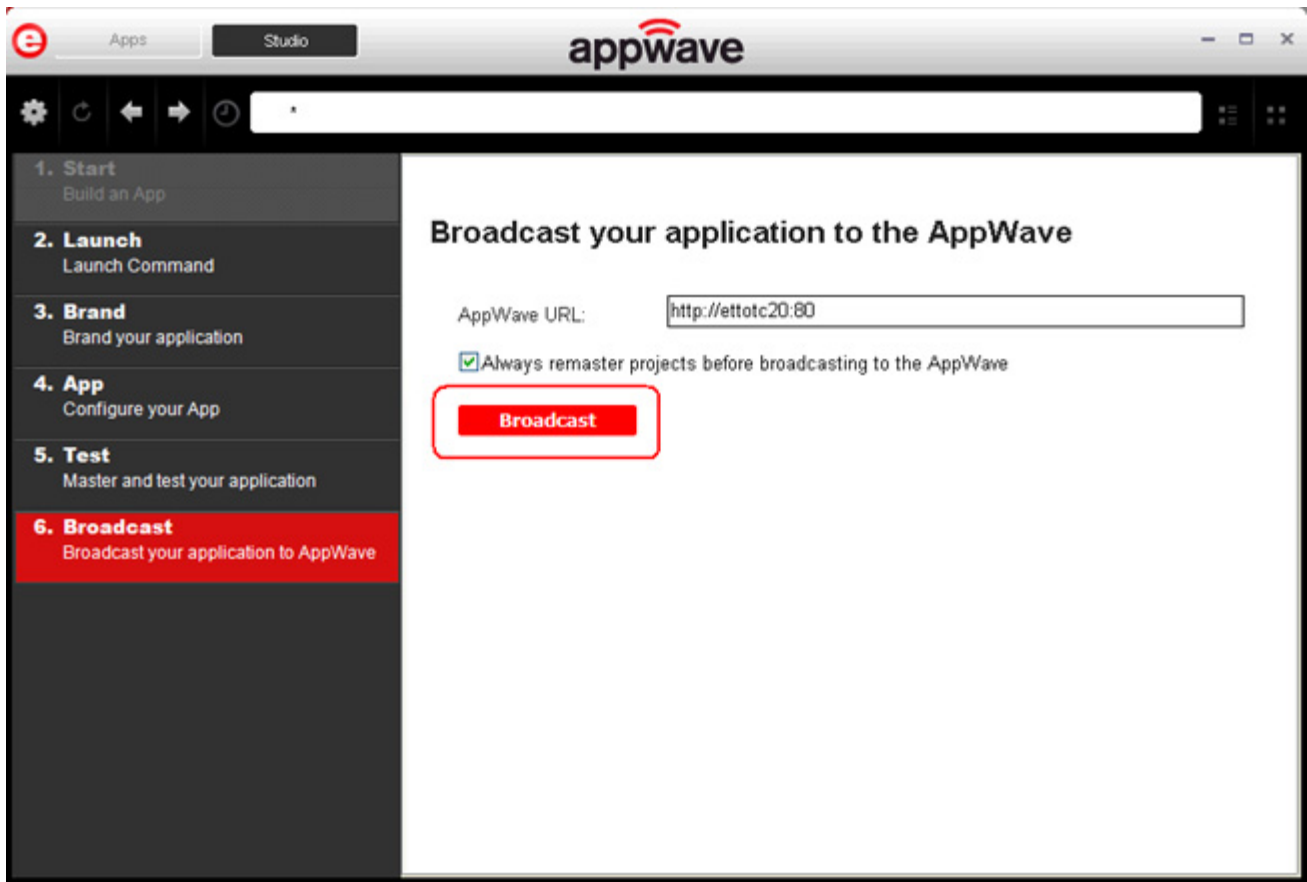
10   A dialog opens asking you to save the changes to your current working file. Click **Yes**, save the file named FileZilla_FTP_Client.tca in the *C:\Documents and Settings\<username>\Embarcadero\AppWaveStudio\Output directory.*

11   Once the file is saved *t*he Broadcast dialog appears.This dialog tracks the process of the broadcasting. Once you get the indication that the broadcast was successful, click **Finished** and the Broadcast dialog closes.

RUNNING THE APP

Once your app is broadcast, the newly mastered app build is made available in the AppWave Browser.

1   Click Apps to return to the AppWave Browser.

2   The following screen shot shows one app has been placed in "KathyB's Folder" designated by the 🔴 , as well as the app that was just mastered.

> **NOTE:**   You can gain access to details and reviews for this App, as well as write reviews.
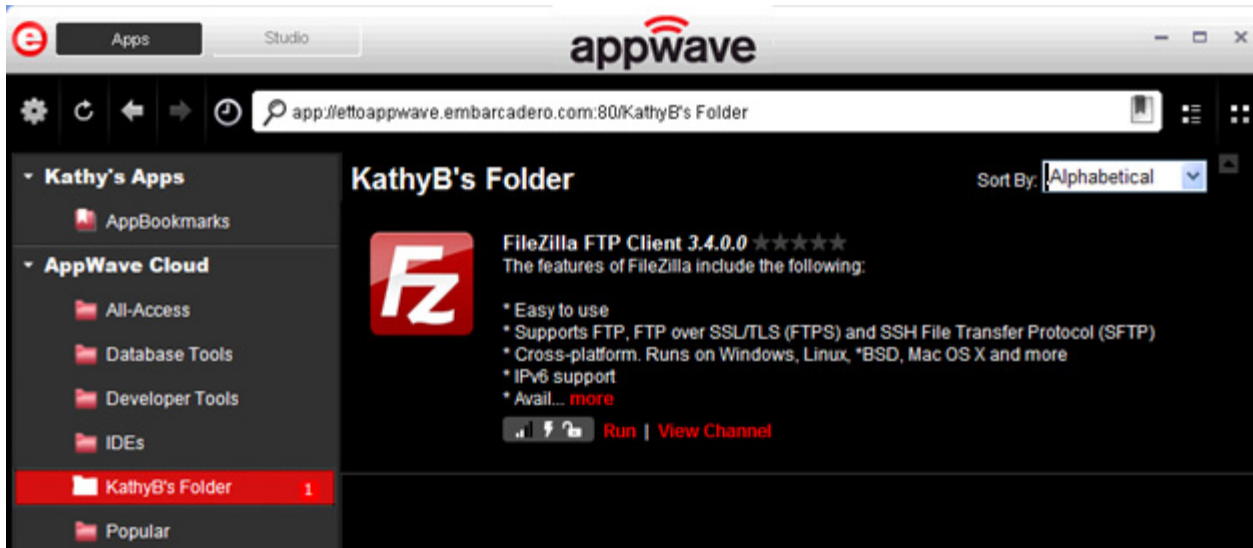
NOTE: A popup also appears in the lower right corner of your screen indicating that new apps are available.
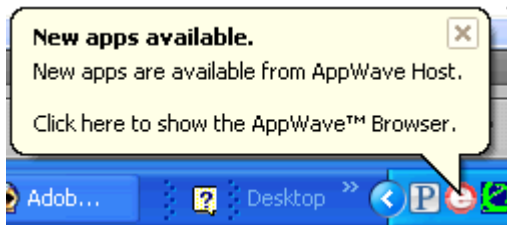
3   Select the app from your folder and click **Run.** If licenses are available for this app the Licensing Information dialog appears which provides the key necessary to complete the final step in the Licensing Automation process.

4   If it is a free third-party app, an End User License Agreement dialog appears.
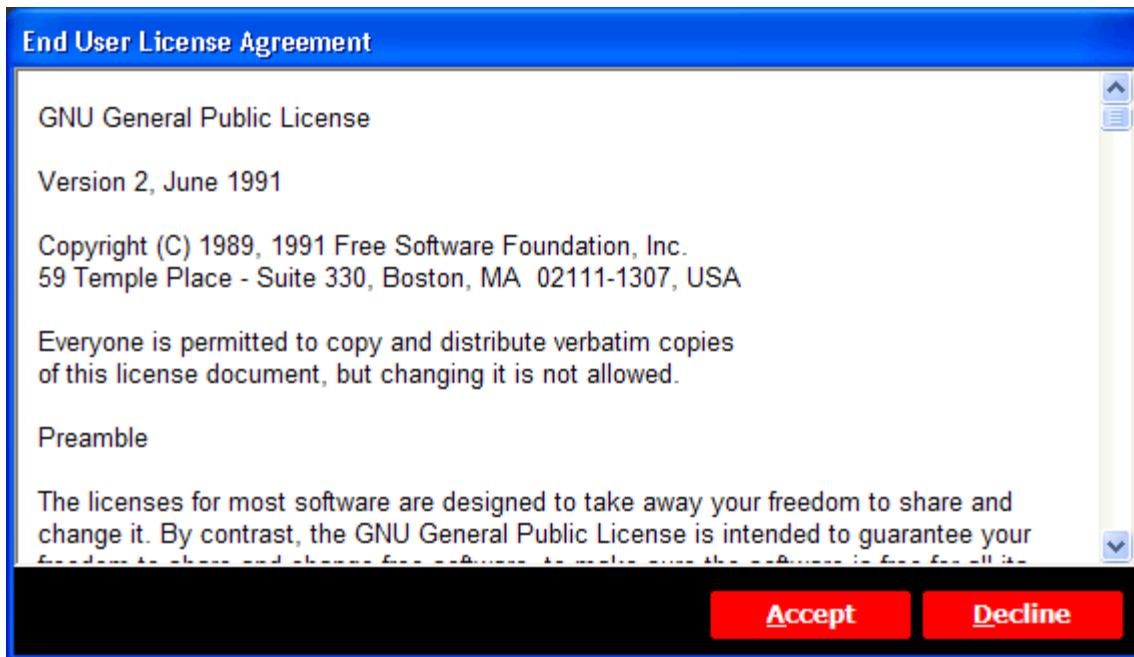
5   Click **Accept** and the app opens.

CAUTION: During mastering, extra services related to the app you are mastering may be launched. At the end of the mastering process, Studio will attempt to shutdown any services it detects, but in some cases Studio cannot detect those services and you must shut them down manually before broadcasting the App.

CREATE APPWAVE SUPPORTED OR CUSTOM APP FROM INSTALL

With this option, you create, edit, and broadcast an app from install using AppWave Studio. The steps to be completed depend on the app you have chosen to install; some steps may be automatically completed for you.

This creation process creates .app and .stp files The .app file uses an MSI Installer which extracts the MSI but does not install the program. The .stp file contains the code and other configuration information that comprises most of the app.

This option is only available through the AppWave Studio. A user must be granted authoring permission by an Administrator.

For more information see: Group Management.

The steps you complete depend on the app you have chosen to install; some steps may be automatically completed for you.

- Start

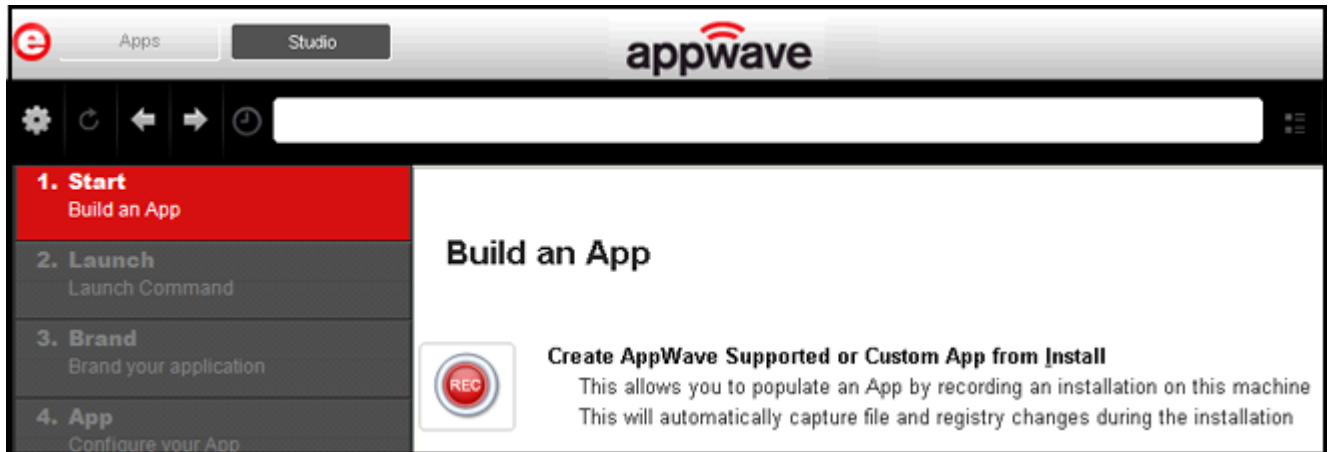- Launch

- [Brand](#)

- [App](#)
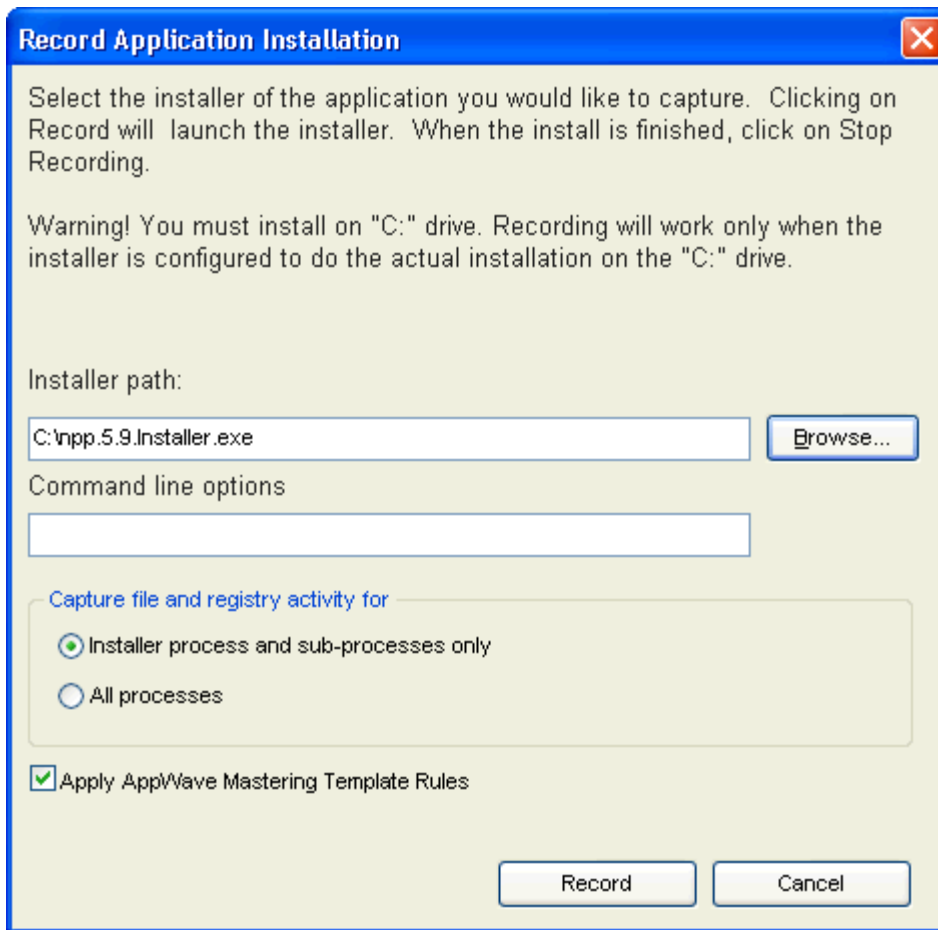
- [Test](#)

- [Broadcast](#)

For more information see [Actions taken in Each Step](#).

START

1   In the AppWave Browser click **Studio** and the **Build an App** page appears with the Start step highlighted in red.

2   Click or the link **Create AppWave Supported or Custom App from Install** and the **Record Application Installation** dialog appears.

3   Browse to the path where the installer file resides. The executable must be on the "C:" drive or the recording will not work. The example above shows the NotePad+ executable

4   In the Command line options field, enter any Command Lines you want to use. For information on commands you can use, see Using the Command Line.

5   In the Capture section you can choose to record only the install process and sub-processes, or all processes. The first option is the default setting.

6   The A**pply AppWave Mastering Template Rules** option is selected by default. As the process goes through the steps to create an App, warnings may be generated indicating missing data based on the template.

7   Click **Record** and the Recording dialog appears (typically in the upper right-hand corner of the screen). The AppWave Studio is minimized during the recording process.

8   Follow the installation wizard for your selected app.

9   The licensing of your app depends on what type of AppWave licensing you have purchased as well as the type of app you are creating. For a full explanation of how to license your app, please refer to: Managing Licensing.

10 Once the install process is complete, click **Finished** in the Recording dialog and the **Record Installation** dialog appears.

> NOTE: Click **Details** to see the details of the installation while it is recording or after it completes. If you want to see it after the recording is complete deselect the "Close on successful completion" option and then click **Details**.

11 Once you click **Finished** in the Record Installation dialog, the next step that opens depends on the app you are installing.

No matter which step you start on, you are then directed to complete the remaining steps to successfully master your app. The different steps are:

- The Launch page is where you select the launch executable and working folder.

- The Brand page is where you provide branding information for the app.

- The App page is where you can configure your app working with files, registry entries, environment variables, and required runtimes.

- The Test page is where you master and test your app before broadcasting to the AppWave.

- The Broadcast page is where you broadcast your mastered app to the AppWave.

For more information, see: Using the Start Task.

LAUNCH

The Launch command is where you select the launch executable.The working folder is then populated. You can also set any parameters needed for app and select an icon that you want to use.

> NOTE: If a Tasks pane is open at the bottom of the page, this means there are fields that are missing required information. Click **Fix** next to the task and your cursor is placed in the appropriate task. Once the tasks are corrected the Tasks dialog closes. The fields with an exclamation mark (⚠) are the fields missing required information.
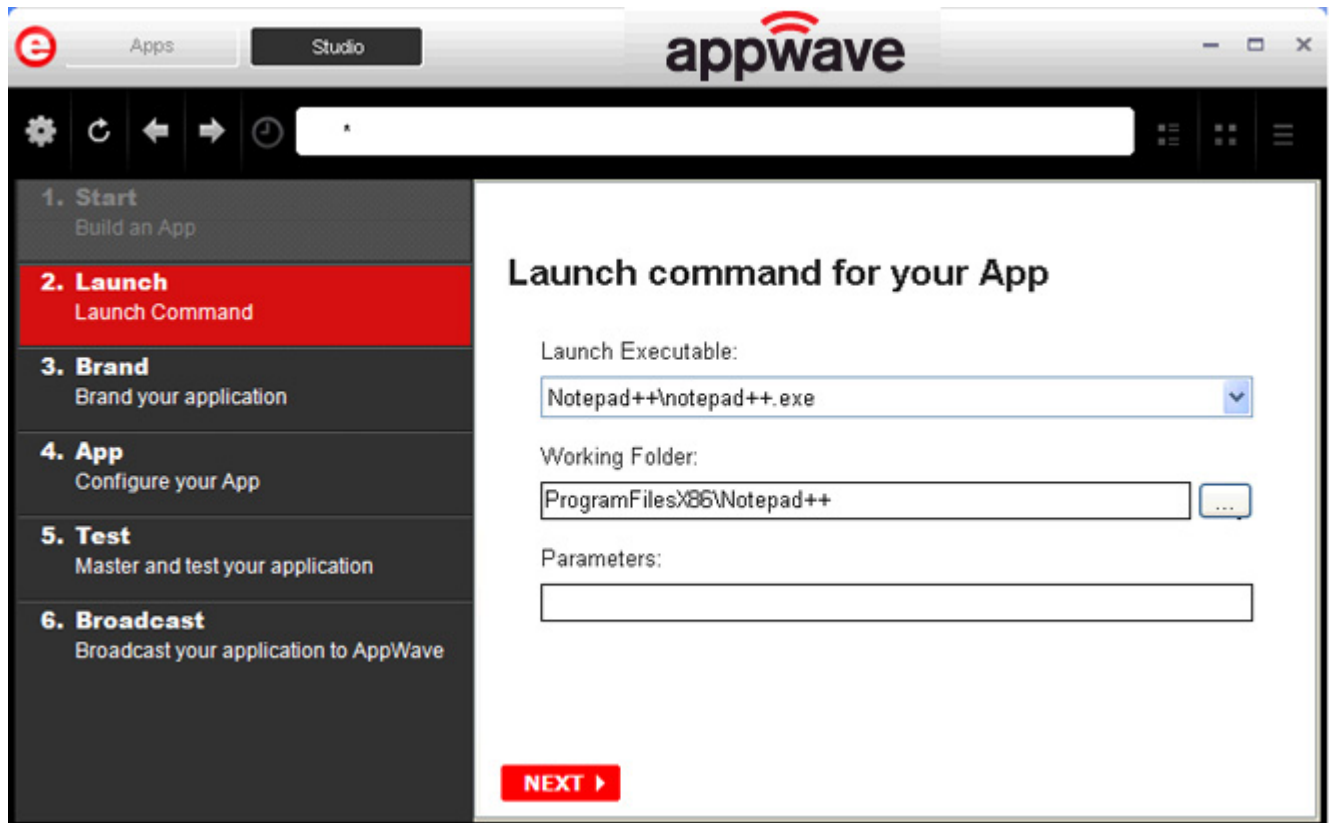
1   Click the drop-down in the Launch Executable and select the executable.

2   Once your select the desired file, the Working Folder is automatically populated.

3   Once the commands are complete click **Next** to advance to the Brand page.

For more information on this step, see: Using the Launch Task.

BRAND

The **Brand** page is where you identify the app. The product information entered here is used in AppWave to identify it.

> **NOTE:** If a Tasks pane is open at the bottom of the page, this means there are fields that are missing required information. Click **Fix** next to the task and your cursor is placed in the appropriate task. Once the tasks are corrected the Tasks dialog closes. The fields with an exclamation mark (⚠) are the fields missing required information.
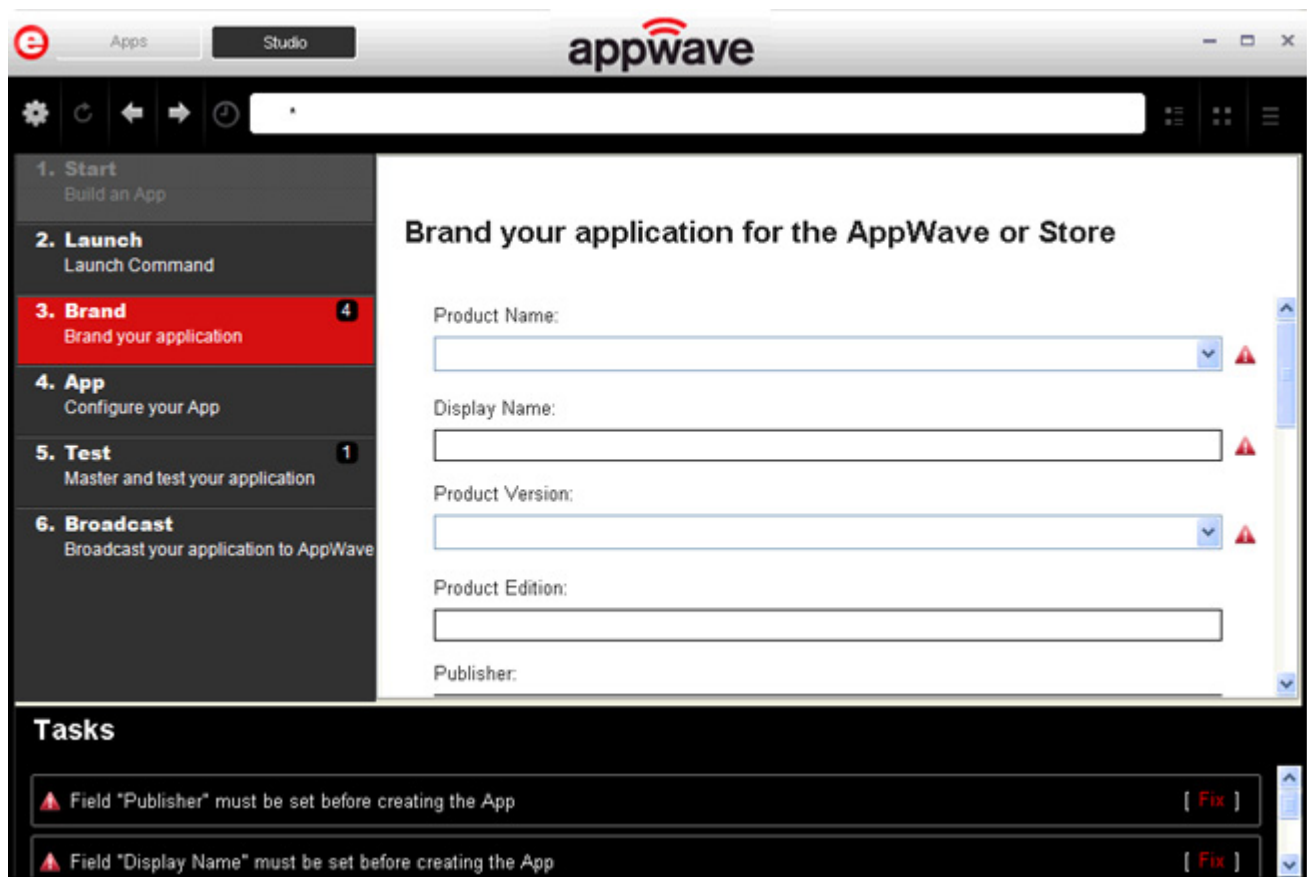

1   Complete the fields on the Brand page. The required fields are: Product Name, Display Name, and Publisher.

2   Click **Next** to advance to the App step.

For more information on each field in this step, see: Using the Brand Task.

App

1   When you have completed the Start, Launch, and Brand steps, click **App** and the **Configure your App** page opens.

    The **Configure your App** page is where you can view what was created when you built a new App. You can edit, add, or delete the files, registry settings, or environment variables that were created during the creation process. You can also set required run times.

2   Click **Files** and the **Files entries for you App** page opens.

3   This displays the working folder which contains the managed files which includes the executable.
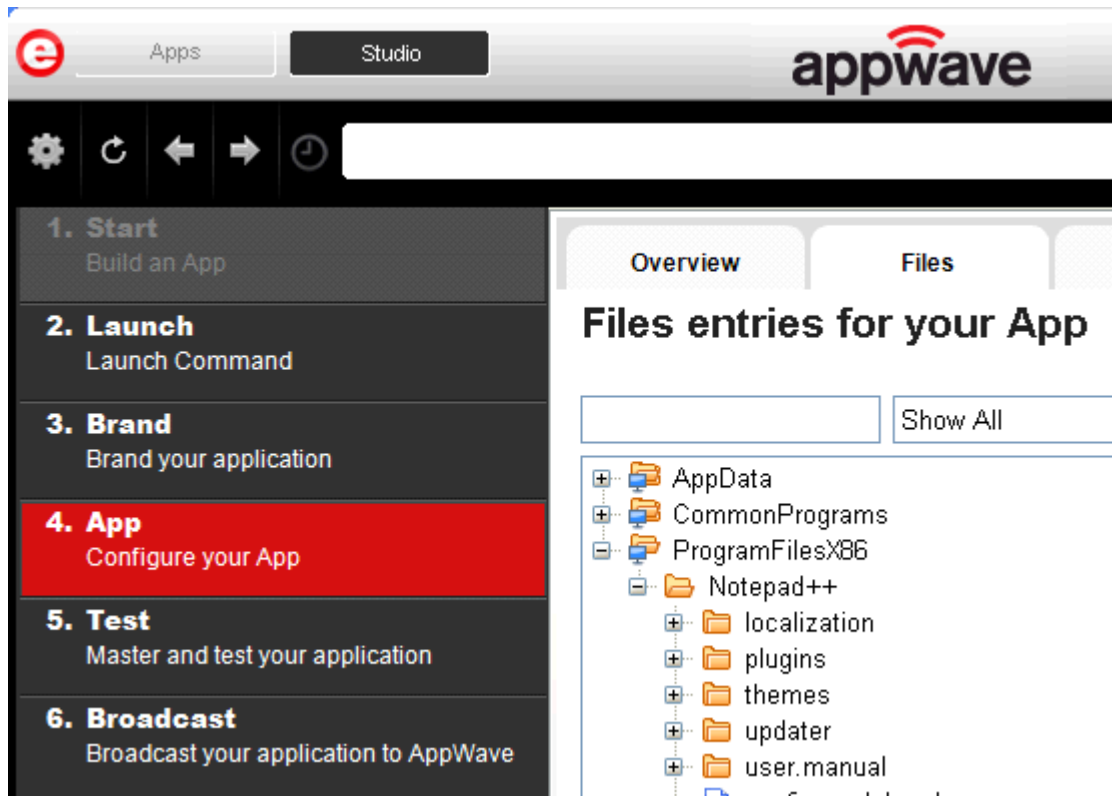
For more information on this step, see: Using the App Task.

Test

Click **Test** to master and test your app locally. This option allows you to master an app file using the given template .tca file.

A .tca file (also known as the Studio file) is a set of information that defines the resources, launch points, and registry settings required to run a product. It also contains information about the product such as the version, name, and editions. The .tca file needs to have the files that goes with it. The .tca file has a file system information that actually list files for the program it needs to master. The files have to be physically present at the directory information that is saved in the .tca data.

1 The Output Folder is automatically populated with the path, i.e., `c:\Documents and Settings\<user>\Embarcadero\AppWaveStudio\Output\`.

2 The File Name is also automatically populated with the information from the Brand page.

CAUTION: Only ASCII characters should be used in File Name field. The mastered app will not launch if non-ASCII characters are used (for example Synérgie).

3 In the Broadcast field if an app file exists for the app (.stp), click edit. Select the App file (.stp) to associate with the .tca file for this app.

> **NOTE:** This .stp file is typically located in the same output directory (`c:\Documents and Settings\<user>\Embarcadero\AppWaveStudio\Output\`).
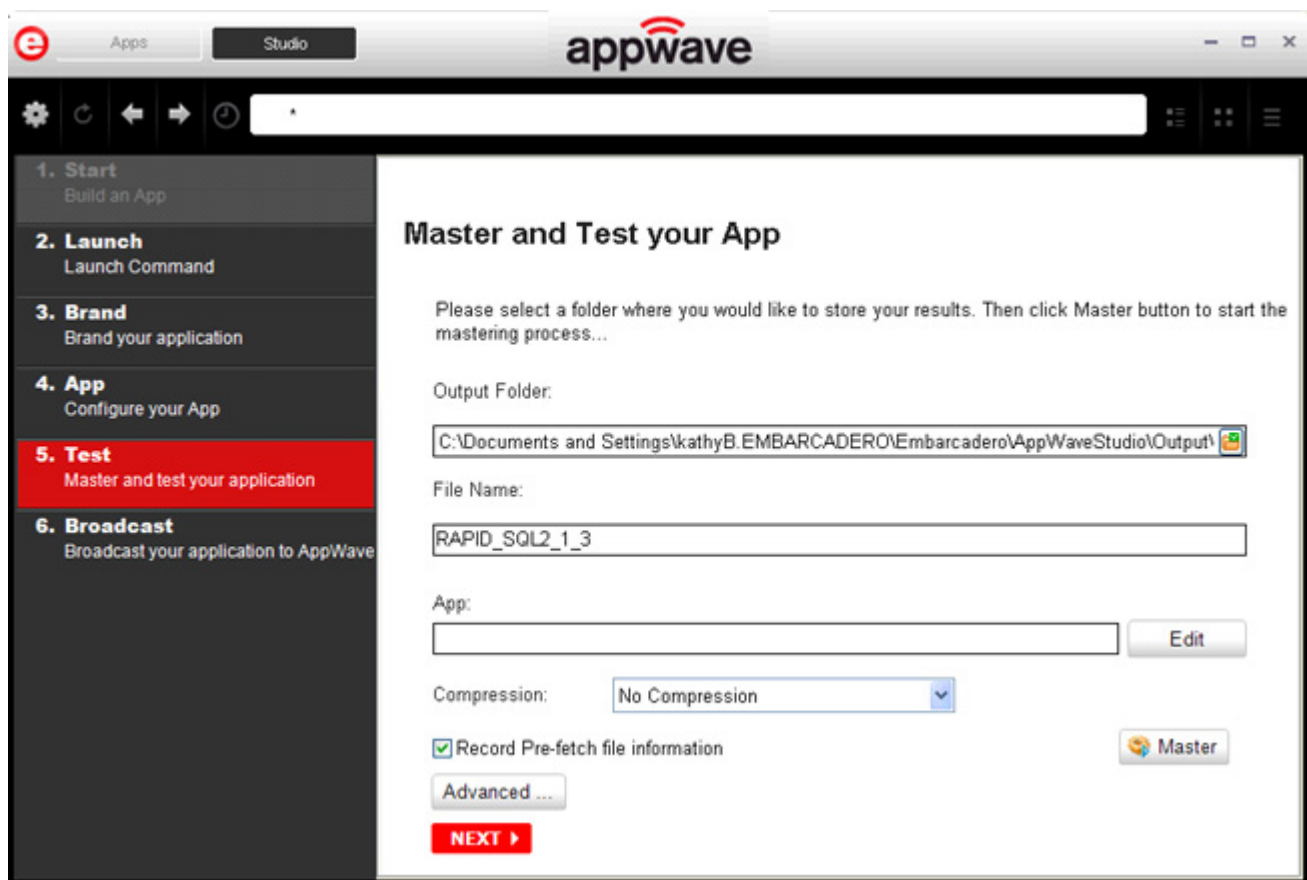
4 Click the drop-down arrow in the Compression field and select the type you want to use. "Best" is the recommended setting for Compression.

5   The Record Prefetch file information is selected by default. For more information on how to create an app with this option, see: Prefetch.

6   If you want to select the type of encryption to use in the app, click Advanced.

7   After your have set and verified your selections, click **Master**, and a confirm dialog opens asking you to save the changes to your current working file. Click **Yes** to continue.

8   The **Master** dialog appears with a progress bar showing the mastering steps. Click **Details** to view the mastering details.

9   Once the mastering is completed you can click **Launch** to open the app or click **Finished** to advance to the Broadcast step.

For more information on this step, see: Using the Test Task.

BROADCAST

From the Test page, click **Next** to advance to the **Broadcast** page.
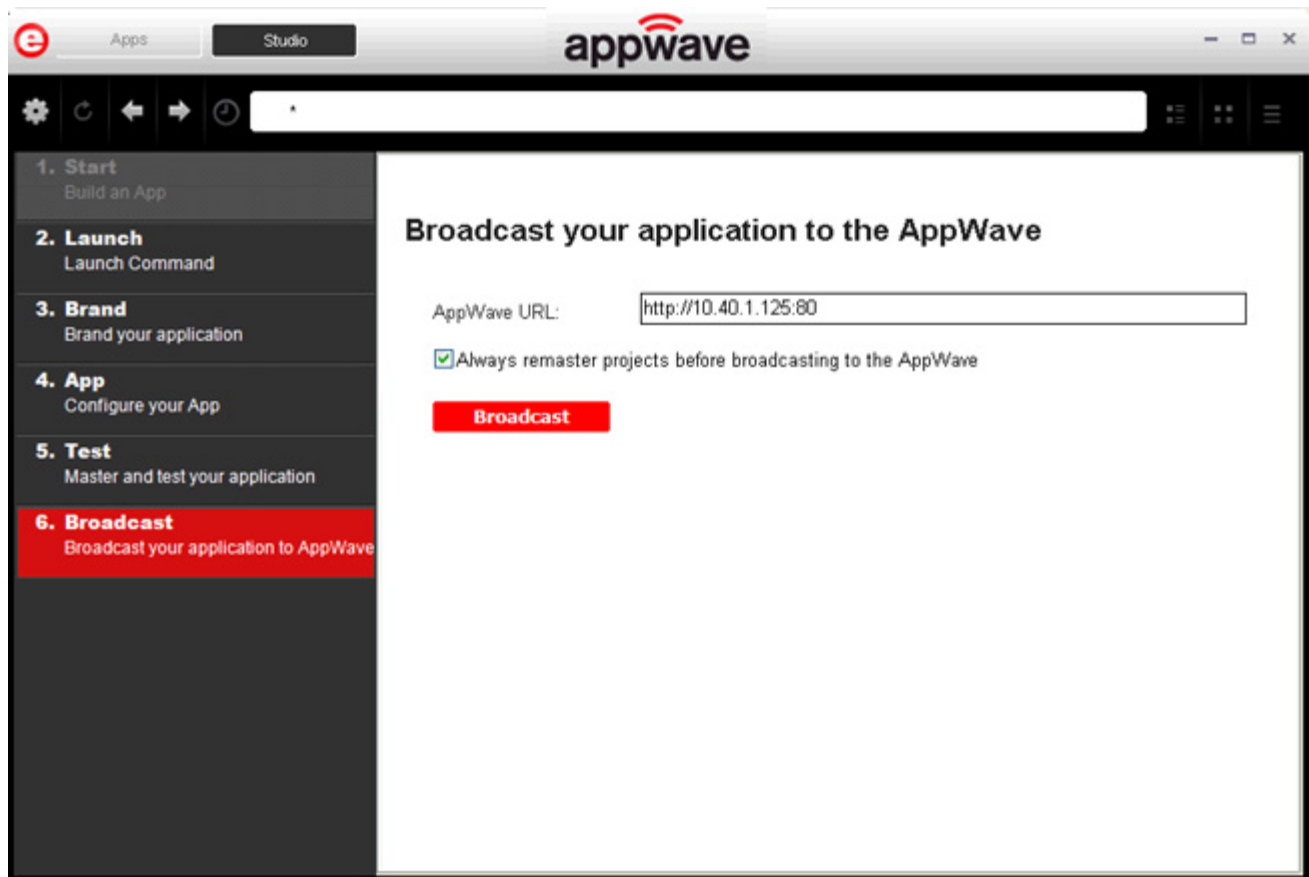
1. The AppWave URL displays the current AppWave.

2. It is recommended that you leave the default setting of **Always remaster projects before broadcasting to the AppWave.** This is to ensure that any changes you have made are included in the mastering.

3. Click **Broadcast** and you are prompted to save the changes to the current working file. Click **Yes** and save your file as a .tca file in a directory of your choice. The default file is
   ```
   c:\Documents and Settings\<user>\Embarcadero\Product
   Repository\<product name>\<product version if applicable>\.
   ```

4. The **Broadcast** dialog opens and shows the steps of the broadcasting processes. It also displays the time it took to complete the operation.

   NOTE: The two files created have the file extensions of .app and .stp and are placed in the C:\Documents and Settings\<user>\Embarcadero\AppWaveStudio\Output folder.

5. Once you get the indication that the broadcast was successful, click **Finished** and the dialog closes.

6   After your app is broadcast, the newly mastered app is placed in a working folder. The following screen shot shows one app has been placed in "KathyB's Folder" designated by the 🔴1️⃣, as well as the app that was just mastered.
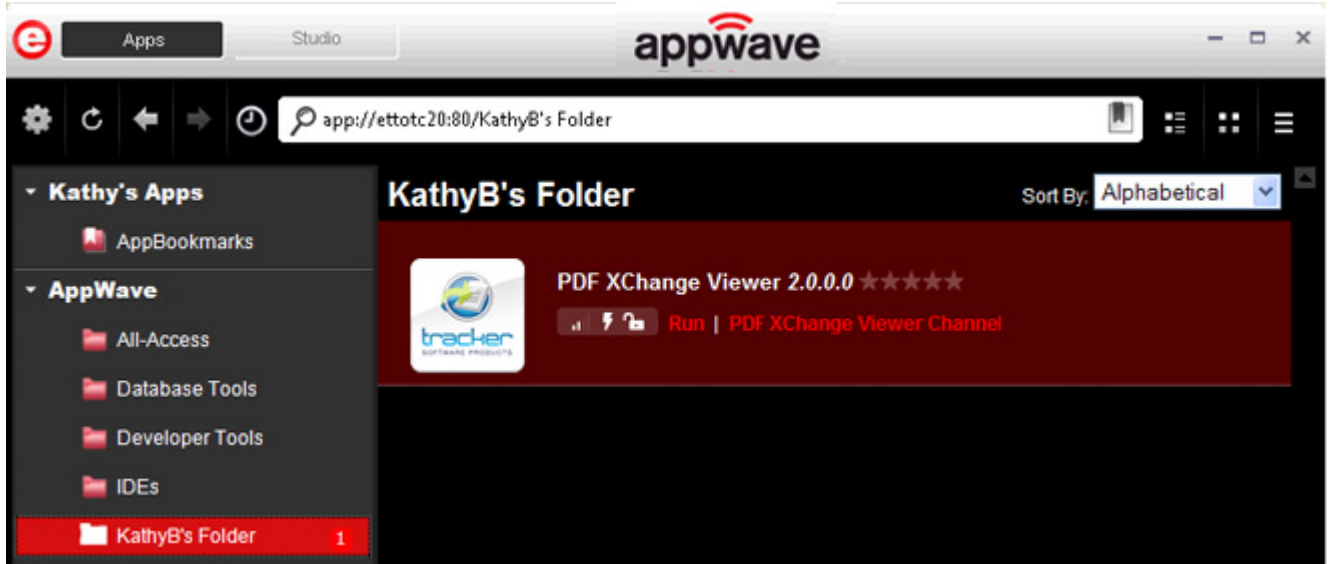
> **NOTE:**   A popup also appears in the lower right corner of your screen indicating that new apps are available.
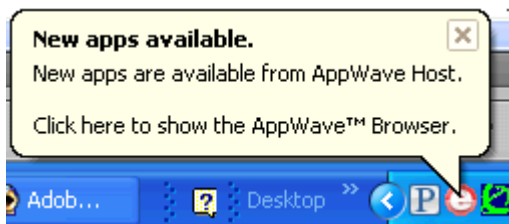
7   Click **Run** and the app opens.

> **NOTE:**   You can gain access to details and reviews for the app. You can also write reviews on the app.

CAUTION: During mastering, extra services related to the app you are mastering may be launched. At the end of the mastering process, Studio will attempt to shutdown any services it detects, but in some cases Studio cannot detect those services and you must shut them down manually before broadcasting the App.

For more information, see: Using the Broadcast Task.

ACTIONS TAKEN IN EACH STEP

Since much of the information needed to complete an app mastering is automatically implemented, each step is listed below with the information that was provided. You have the option to click on a step and provide missing information (for example a product description in the Brand step).

> **NOTE:** The following information is from the creation the FileZilla App.

Launch Command for your App

Field	Information Provided:
Launch Executable	FileZilla FTP Client\filezilla.exe
Working Folder	ProgramFilesX86\FileZilla FTP Client
Parameters	None Provided
Icon	C:\Documents and Settings\kathyB.EMBARCADERO\Embarcadero\AppWaveStudio\Output\filezilla_ftp_client.ico

Brand your Application for the AppWave

Field	Information
Product Name	FileZilla FTP Client
Display Name	FileZilla FTP Client
Product Version	3.4.0.0
Product Edition	None Provided
Publisher	FileZilla Project
EULA	C:\Documents and Settings\kathyB.EMBARCADERO\Embarcadero\AppWaveStudio\Output\FileZilla_FTP_ClientAppWave.htm
Screen Shot	C:\Documents and Settings\kathyB.EMBARCADERO\Embarcadero\AppWaveStudio\Output\FileZilla_FTP_ClientAppWave.png
Web Site	http://filezilla-project.org
Support Contacts	None Provided
System Requirements	None Provided
Product Description	FileZilla Client is a fast and reliable cross-platform FTP, FTPS and SFTP client with lots of useful features and an intuitive graphical user interface.

Field	Information

Long Product Description	The features of FileZilla include the following:
	  * Easy to use
	  * Supports FTP, FTP over SSL/TLS (FTPS) and SSH File Transfer Protocol (SFTP)

Configure Your App

Division	Description
File Entries	There are 472 files, 0 are exposed to the host machine
Registry Entries	Registry entries provided in the following references: HKEY_CLASSES_ROOT; and HKEY_LOCAL_MACHINE
Environment Variables	None
Runtime	.Net 3.5 runtime plug-in, Adobe Flash runtime plug-in, Microsoft J# runtime plug-in, and Microsoft Visual C++ 2005 SP1 redistributable runtime plug-in

Master and Test Your App

Field	Information
Output Folder	C:\Documents and Settings\<user>\Embarcadero\AppWaveStudio\Output\
File Name	FileZilla_FTP_Client3_4_0_0
App	C:\Documents and Settings\kathyB.EMBARCADERO\Embarcadero\AppWaveStudio\Output\FileZilla_FTP_Client3_4_0_0.stp
Compression	No Compression
Record Pre-fetch file information	Selected

Broadcast your Application to the AppWave

Field	Information
AppWave URL	Your default url address.
Always remaster projects before broadcasting to the AppWave	Selected

CREATE APPWAVE SUPPORTED OR CUSTOM APP FROM SCRATCH

This section walks you through how to create an app from an application's executable file. This is a custom application that does not have the mastering support. This means that you need to manually insert Product Name, Display Name, Version, Long Description, etc. Once your product is mastered and placed in your AppWave, you won't need to master it again.

This option is only available through the AppWave Studio. A user must be granted authoring permission by an Administrator.

For more information see: Group Management.

This option provides a way for you to create an app file from a given installer file. It actually extracts all the install files from the installer, and then puts these extracted files in a directory under the OutputLocation directory.

The steps to be completed are:

- Start
- Launch
- Brand
- App
- Test
- Broadcast

START

1   In the AppWave Browser click **Studio** and the **Build an App** page appears with first step highlighted in red.
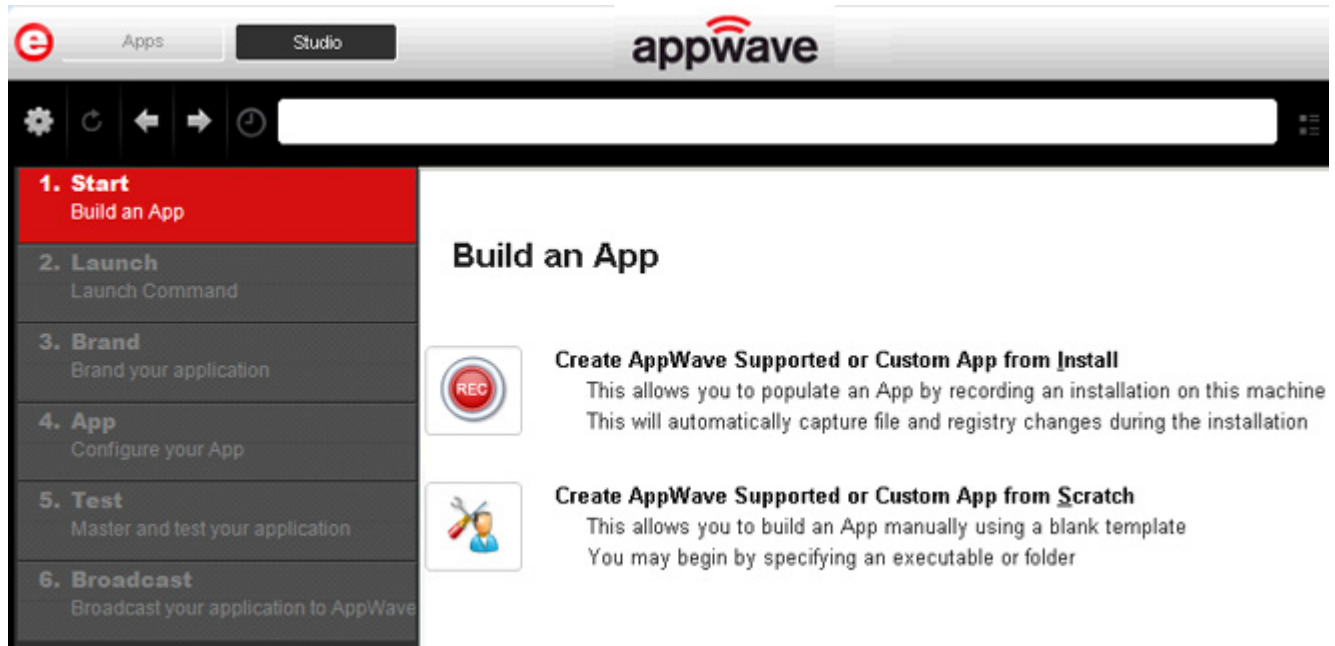
2   Click the **Create AppWave Supported or Custom App from Scratch** link or click   .

1   The **Select Folders and FLiles to include** dialog opens. For the **Initial Folder Name**, enter an existing folder name or create a new one. This is the "working folder" for your app.

2   In **Select a file**, browse to the Installer path and select the executable. The Installer path must on the "C:" drive. For example: C:\PDFViewer.exe.

3   In **Select a folder** browse to an existing folder or make a new folder. This will then contain the executable and files from the creation process.

4   Click **OK** and the **Create From Scratch** dialog appears. You can select the option to close this dialog on successful completion of the operation. The time it took to complete the operation is displayed.
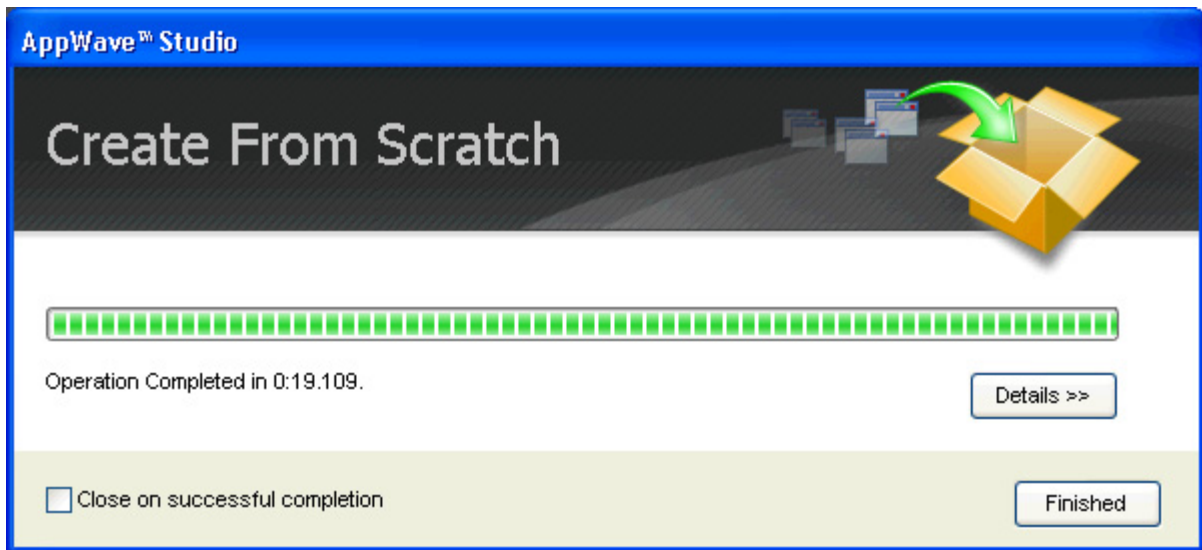
NOTE: Click **Details** to see the details of the operation while it is creating or after it completes. If you want to see it after the operation is complete deselect the "Close on successful completion" option and then click **Details**.

5   Depending on the option you selected, the Create from Scratch dialog may or may not automatically close. If you selected the "Close on Successful Completion either the <u>Launch</u> or <u>Brand</u> page automatically appear, depending on the information provided by the executable. If you did not select that option, click **Finished** to advance to the Launch page.

For more information on this first step, see: <u>Using the Start Task</u>.

LAUNCH

The Launch command is where you select the launch executable. The working folder is then populated. You can also set any parameters needed for app.
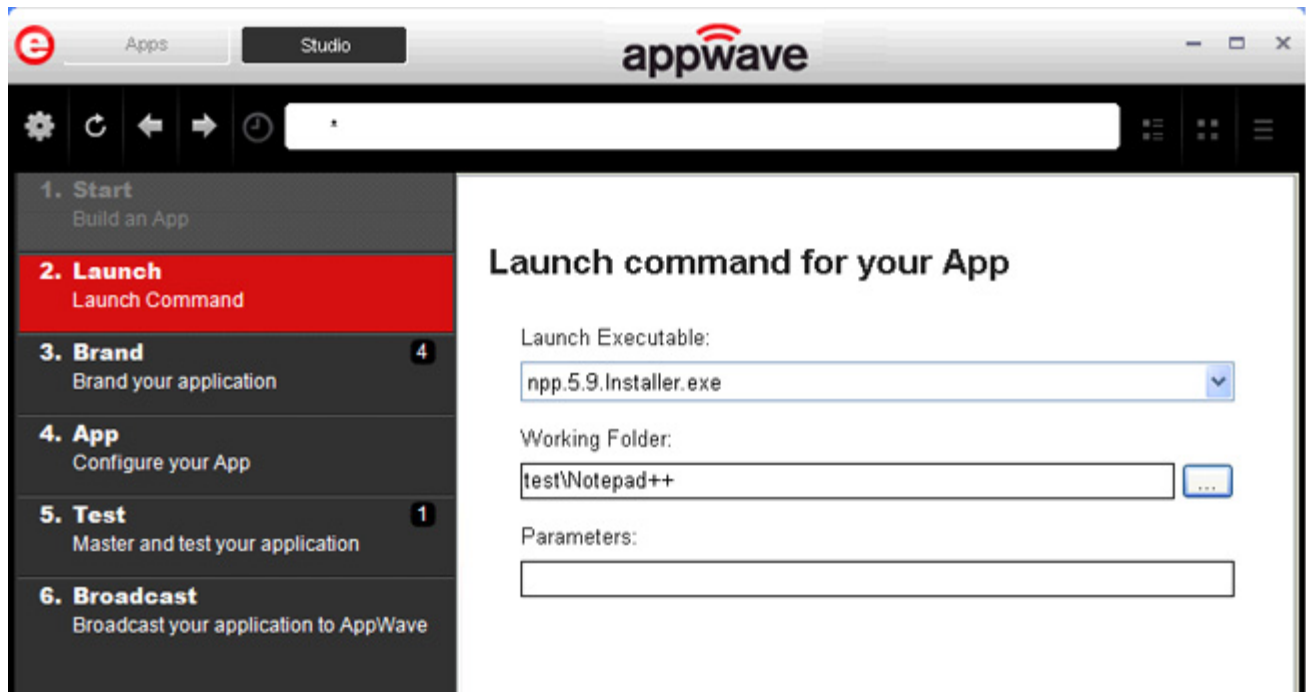
1   Click the drop-down arrow in the Launch Executable and select the executable.

>   **NOTE:** If a Tasks pane is open at the bottom of the page, this means there are fields that are missing required information. Click **Fix** next to the task and your cursor is placed in the appropriate task. Once the tasks are corrected the Tasks dialog closes. The fields with an exclamation mark (⚠) are the required fields missing information.

2   Once your select the desired file, the Working Folder is automatically populated. The name of the working folder is the folder name you entered in the Initial Folder Name field in the first step.

3   Once the commands are completed click **Next** or Brand.

For more information on this step, see Using the Launch Task.

BRAND

There is where you enter critical information for your app. This information for the mastered app appears in the Apps tab of Browser.
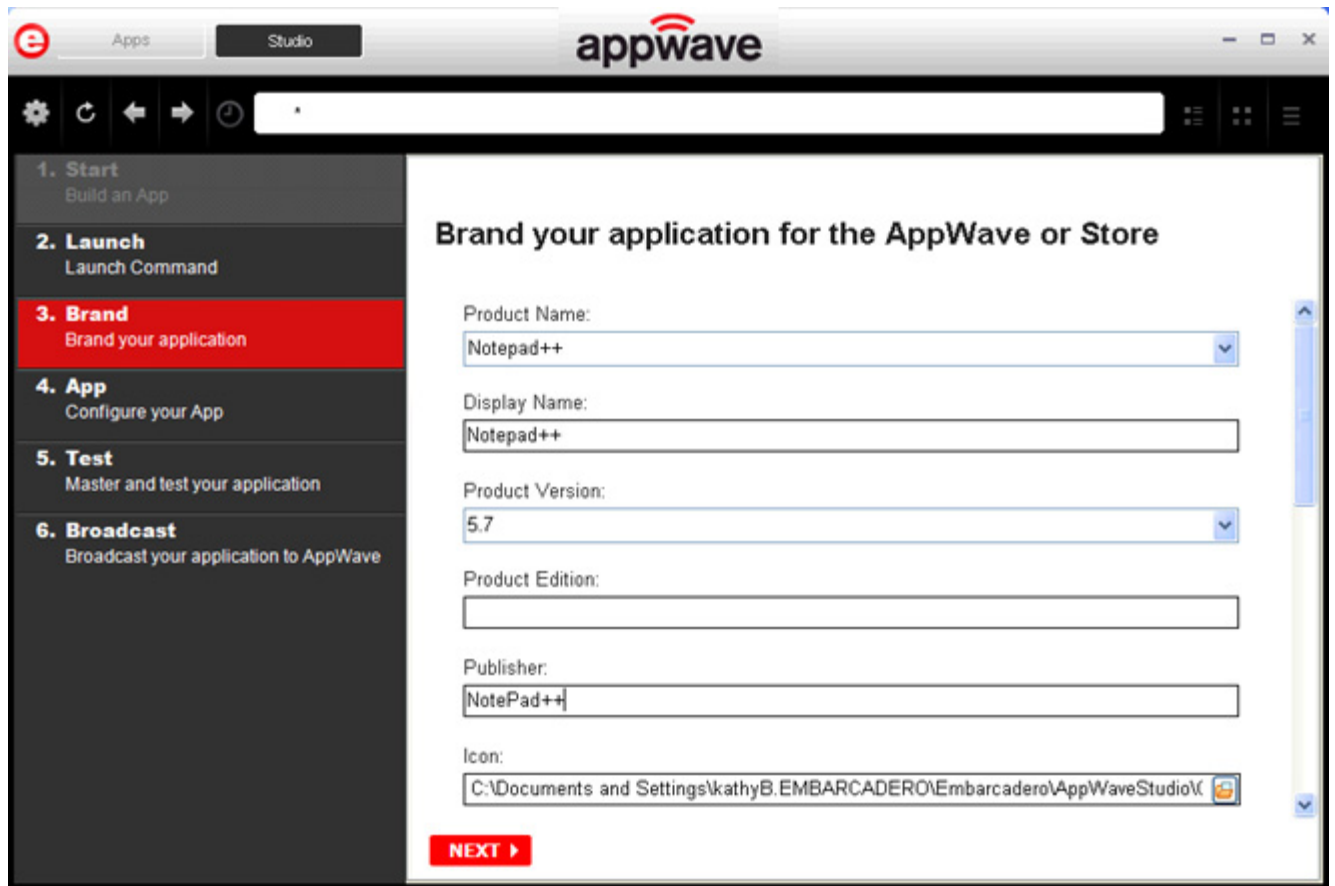
NOTE: If a Tasks pane is open at the bottom of the page, this means there are fields that are missing required information. Click **Fix** next to the task and your cursor is placed in the appropriate task. Once the tasks are corrected the Tasks dialog closes. The fields with an exclamation mark (⚠) are the required fields with missing information.

1   Complete the fields on the Brand page and click **Next** to advanced to App. The required fields are: Product Name, Display Name, and Publisher.

2   If applicable, the Icon field is also automatically populated. If not, you can select the icon file (.ico) from the Open dialog.

3   Click **Next** to advanced to the App page or click the App step listed on the left side of the page.

For more information on each field on this page, see: Using the Brand Task.

APP

1 When you have completed the Start, Launch, and Brand steps, click **App** and the **Configure your App** page opens.

   This is where you can view what was created when you built a new App. You can edit, add, or delete the files, registry settings, and environment variables that were created during the creation process. You can also set required run times.

2 Click **Files** and the **Files entries for you App** page opens.

3 The folder you created or selected in the first step is shown. Once the managed file is broadcast it will also be shown.
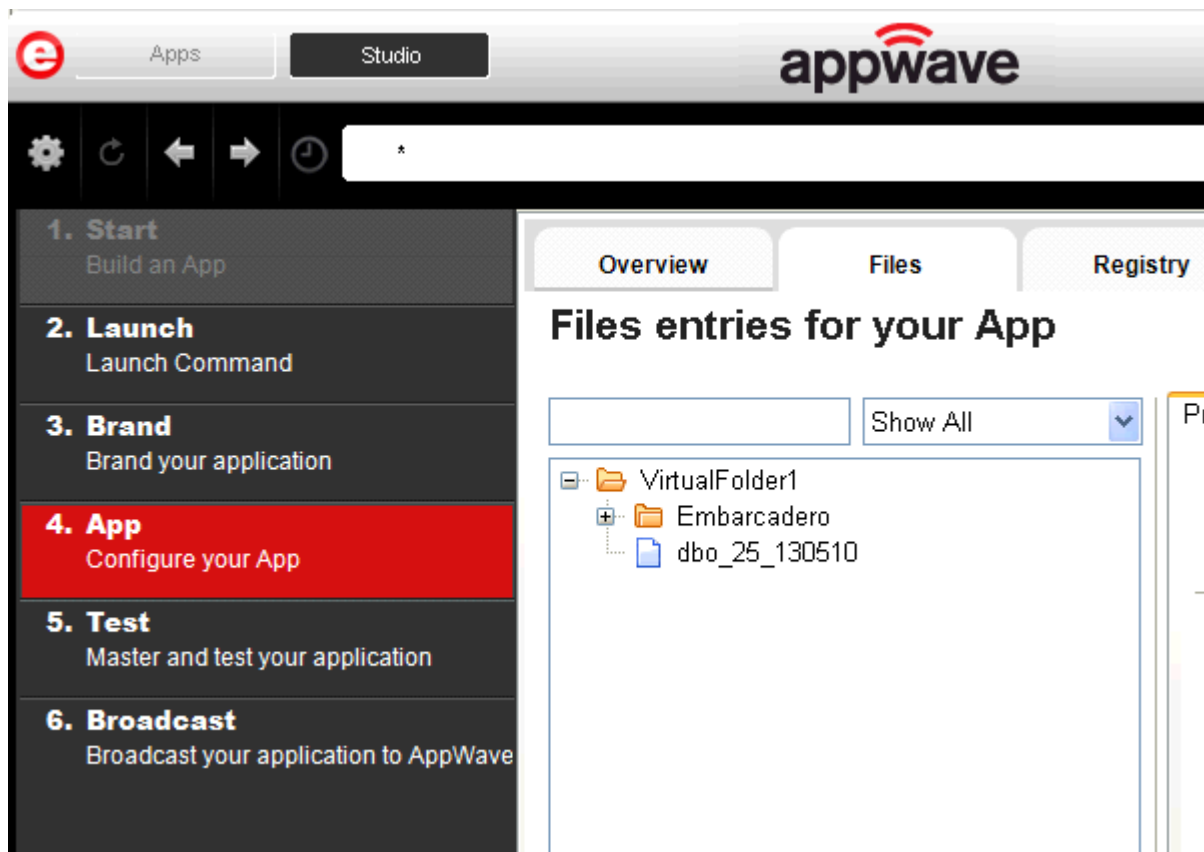
4 Click **Next** to advance to the **Registry entries for your App**. All registries and any changes to those registries are listed.
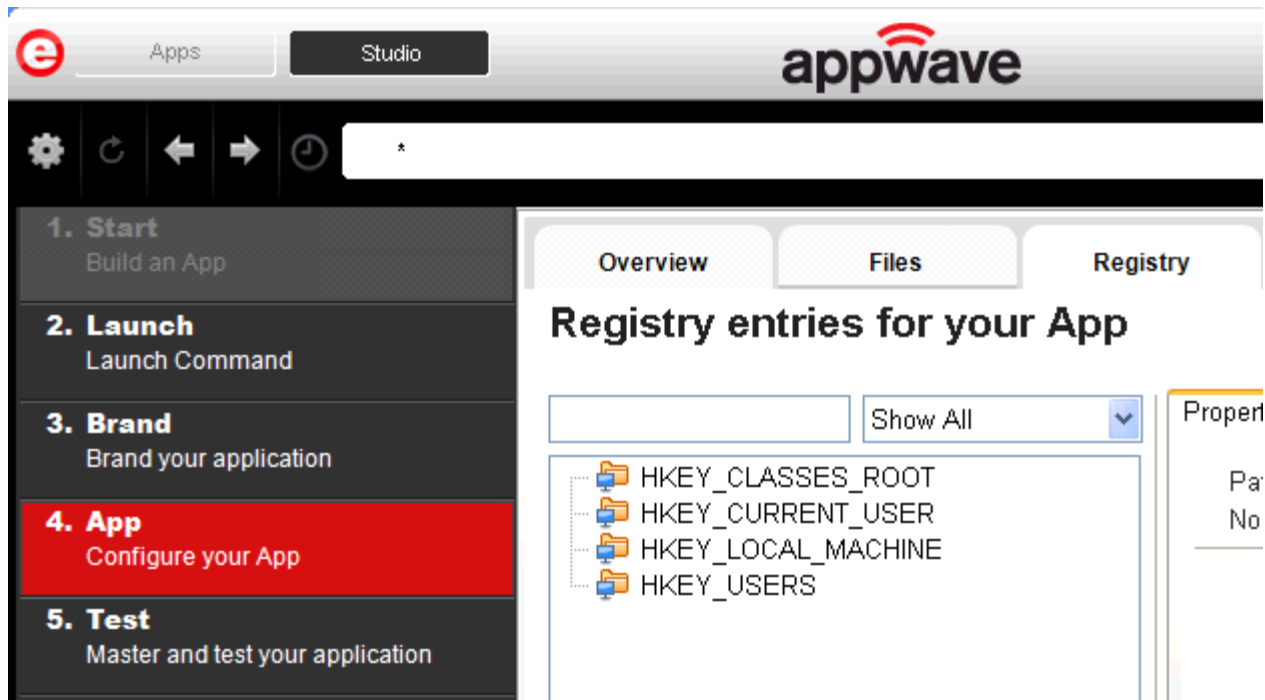
5 Click **Next** and the **Environment variables for your App** appears. You can add, edit or remove any Environment variables on this page.

6 Click **Next** to advance to the **Runtimes required by your App** page. A default list of runtime plug-ins are listed. You can select the ones required by your app.
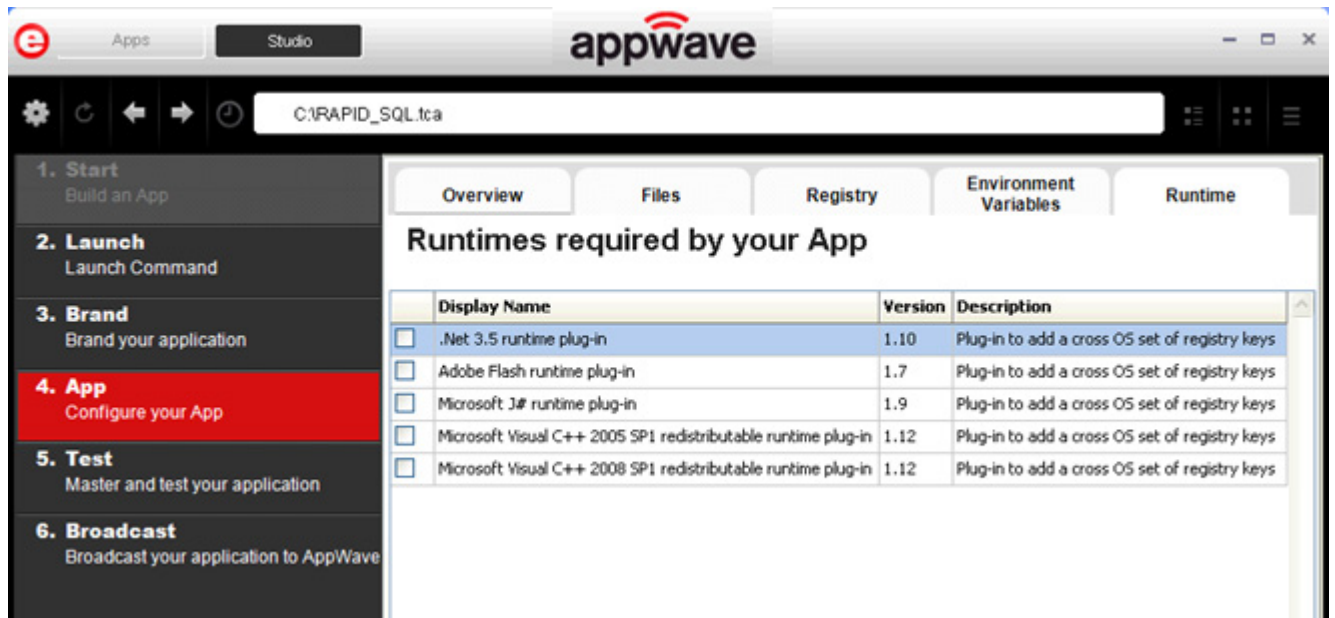
For more information on this step, see: Using the App Task.

TEST

Click **Test** to master and test your app locally. This option allows you to master an app file using the given template .tca file and all the associated files. This is a file system that actually list files for the program it needs to master. The files have to be physically present at the directory information that is saved in the .tca data.

The mastering process verifies the files that are listed in the .tca file data. It then uses these files to complete the mastering of the app file. The template .tca file needs to be a complete and valid .tca file with all the required data (i.e. Publisher, Vendor, etc.) or the mastering will fail. The result of this process is the creation of an App OutputName .stp file.
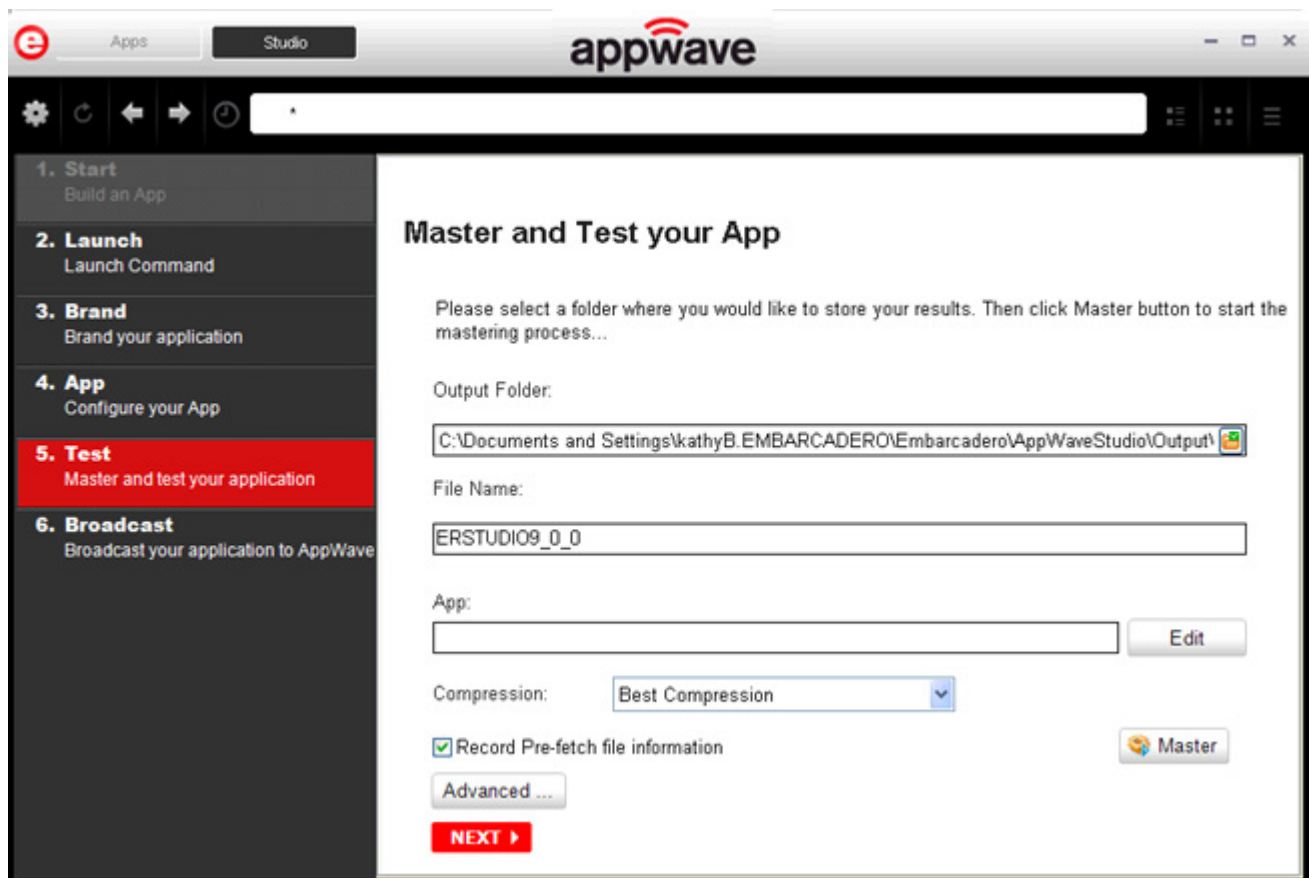
1   The Output Folder is automatically populated with the path.

2   The File Name is also automatically populated with the information from the Brand page.

CAUTION: Only ASCII characters should be used in File Name field. The mastered app will not launch if non-ASCII characters are used (for example Synérgie).

3   In the App field, if an app file exists for the app (.stp), click edit to open the **Select a Mastered App** dialog. Select the App file associated with the .tca file for this app.

4   Click the drop-down arrow in the Compression field and select the type you want to use. "Best" is the recommended setting for Compression.

5   The **Record Prefetch file information** is selected by default. For more information, see: [Prefetch](#).

6   If you want to select the type of encryption to use in the app, click [Advanced](#).

7   After your have set and verified your selections, click **Master**, and a confirm dialog opens asking you to save the changes to your current working file. Click **Yes** and save your file in the Save As dialog.

8   The **Master** dialog appears with a progress bar showing the mastering steps. The time it took to complete the operation displays at the end of the mastering.

9   Click **Details** to view the mastering details if you desire.

10   Once the mastering is completed you can click **Launch** to open the app or click **Finished** to advance to the Broadcast step.

For more information on this step, see: [Using the Test Task](#).

BROADCAST

Click **Broadcast** to complete the last step in the sequence. This step makes the app available in the AppWave Browser.

CAUTION: During mastering, extra services related to the app you are mastering may be launched. At the end of the mastering process, Studio will attempt to shutdown any services it detects, but in some cases Studio cannot detect those services and you must shut them down manually before broadcasting the App.
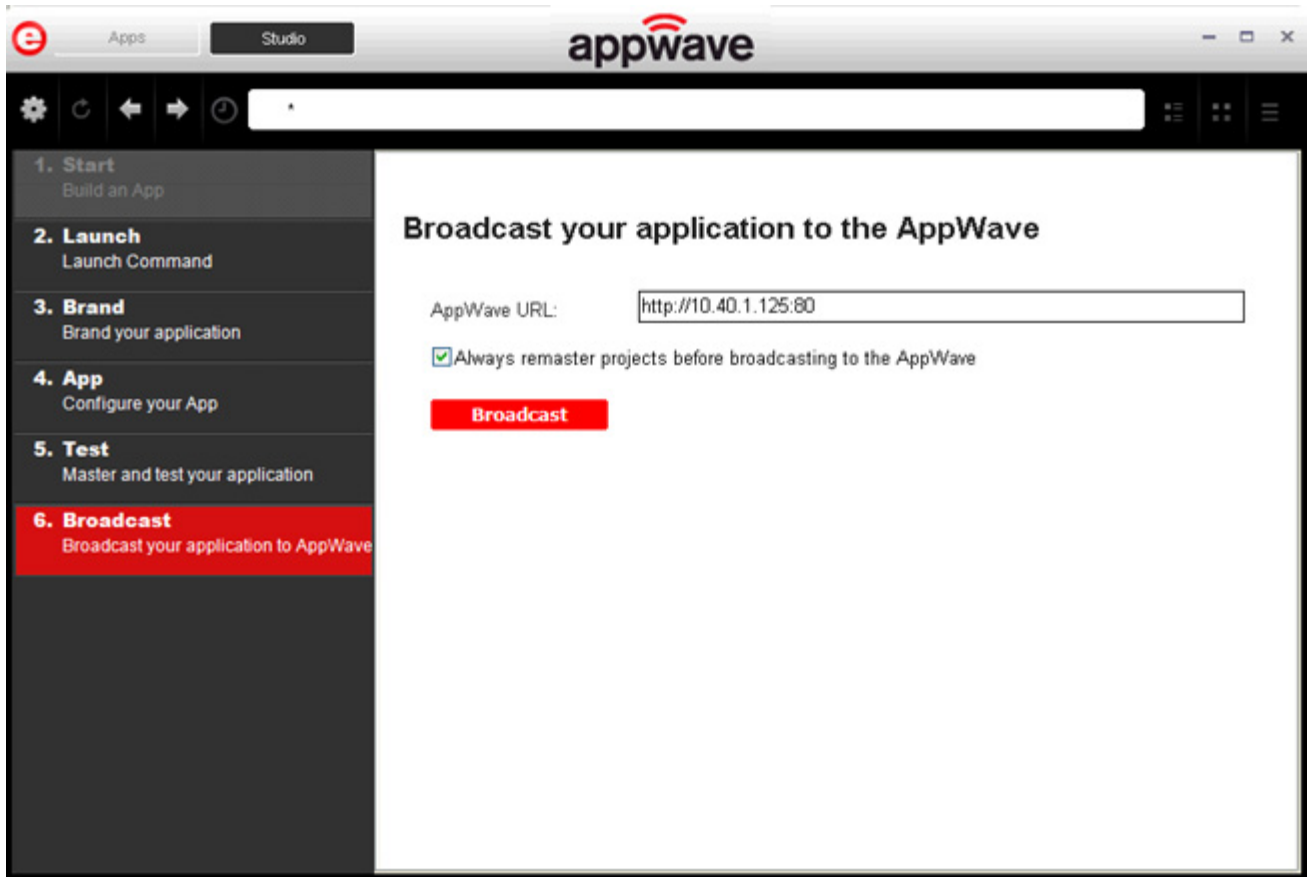
1   The default AppWave URL is listed.

2   By default the **Always remaster projects before broadcasting to the AppWave** is selected. This ensures that any changes you have made are included in the master.

3   Click **Broadcast** and if you haven't saved your file yet, you are advised that you must save the changes to your current working file before broadcasting.

   NOTE: If a .stp file already exists for the app you want to broadcast you get the following message:
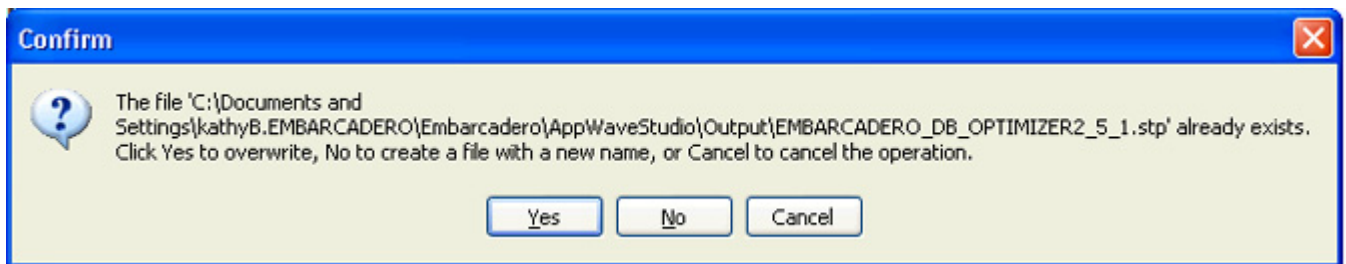
   NOTE: If you try to broadcast an app that has the same name, version, and edition of an existing App, you get the following message:
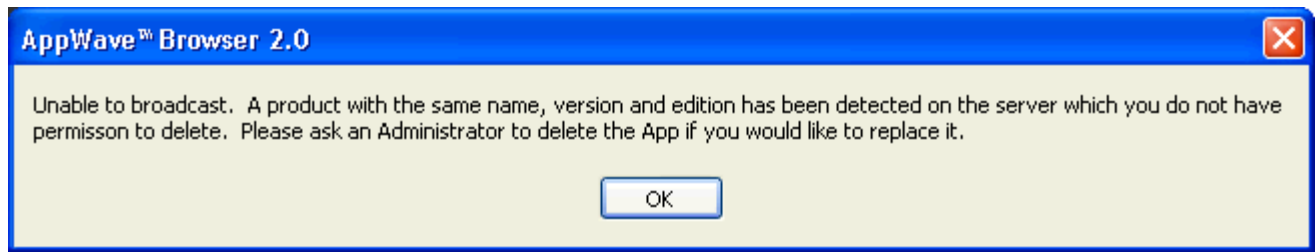
4   Once you have saved your file and closed the Save As dialog, the Broadcast dialog appears. The time it took to complete the operation is displayed at the end of the broadcast.

5   Click **Finished** and the app is ready to be broadcast.

   NOTE: Once the broadcasting is complete, a message window appears in the lower right-hand corner telling you that New apps are available from the AppWave Host.
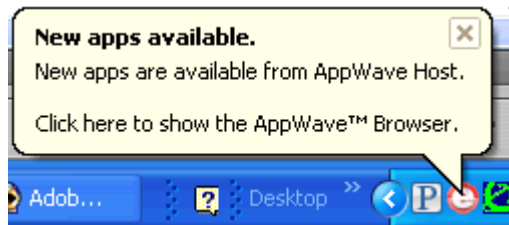
For more information on this step, see: Using the Broadcast Task

USING THE APP

1   Once you have created the app using AppWave Studio, it is available on the AppWave Browser.

2   Click **Apps** and select the **AppWave** folder.

3   During the Broadcast process, if a folder doesn't already exist, a folder with your user name (for example "Kathyb's Folder") is created in the AppWave folder.
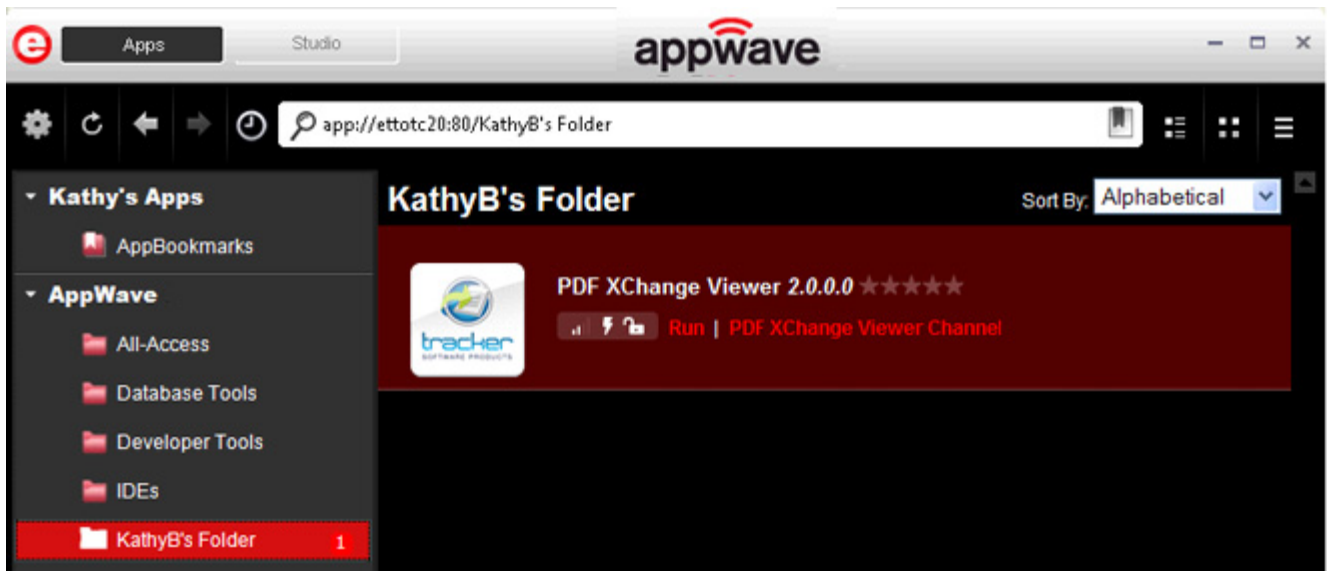
4   Click **Run** to start the app.

> **NOTE:**   For more information on the Apps portion of the app, please see the *AppWave Browser Apps User Guide*.

CREATE AN APP WITH MULTIPLE VERSIONS

When you master an app using an AppWave Template, that template may contain multiple versions of the app. If that is the case, you have the option to select one of the versions listed, and then either open the app that is already installed or record the version listed.

1   In **Apps,** search (   ) for "AppWave Template" and all the apps with that designation are listed. This template is indicated by an icon (**T**) located to the right of the app icon.

2   Scroll down and select the app that you want to master. In the following example we scrolled down to "NotePad++" which has an AppWave Template with multiple versions.
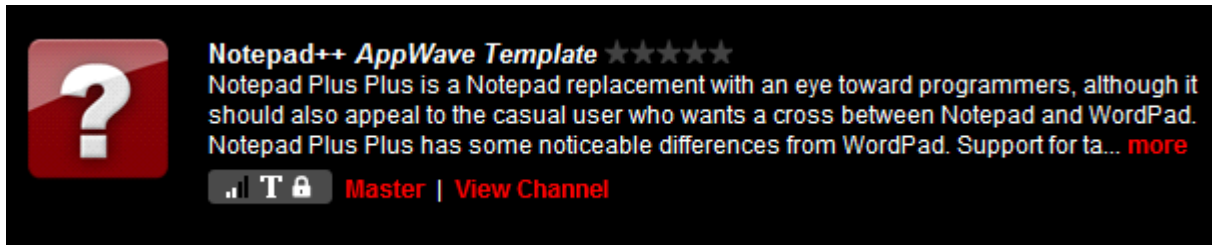
3   Click **Master** located to the right of the icon. A Record Installation dialog briefly appears.

4   Also, an indicator window opens in the lower right-hand corner of your screen showing that the selected file has finished streaming. The file extension .aw indicates it is the AppWave Template file.
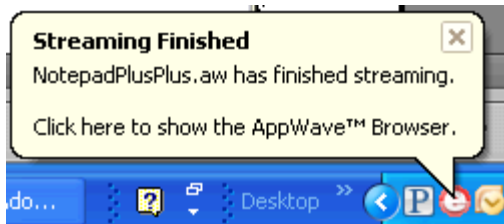
In Studio, the first step of the mastering process opens. However since the app you want to master has different versions included in template, you are given of choice of which version you want to work with in the Build an App page. You can also choose to open an existing installation or record a new one.
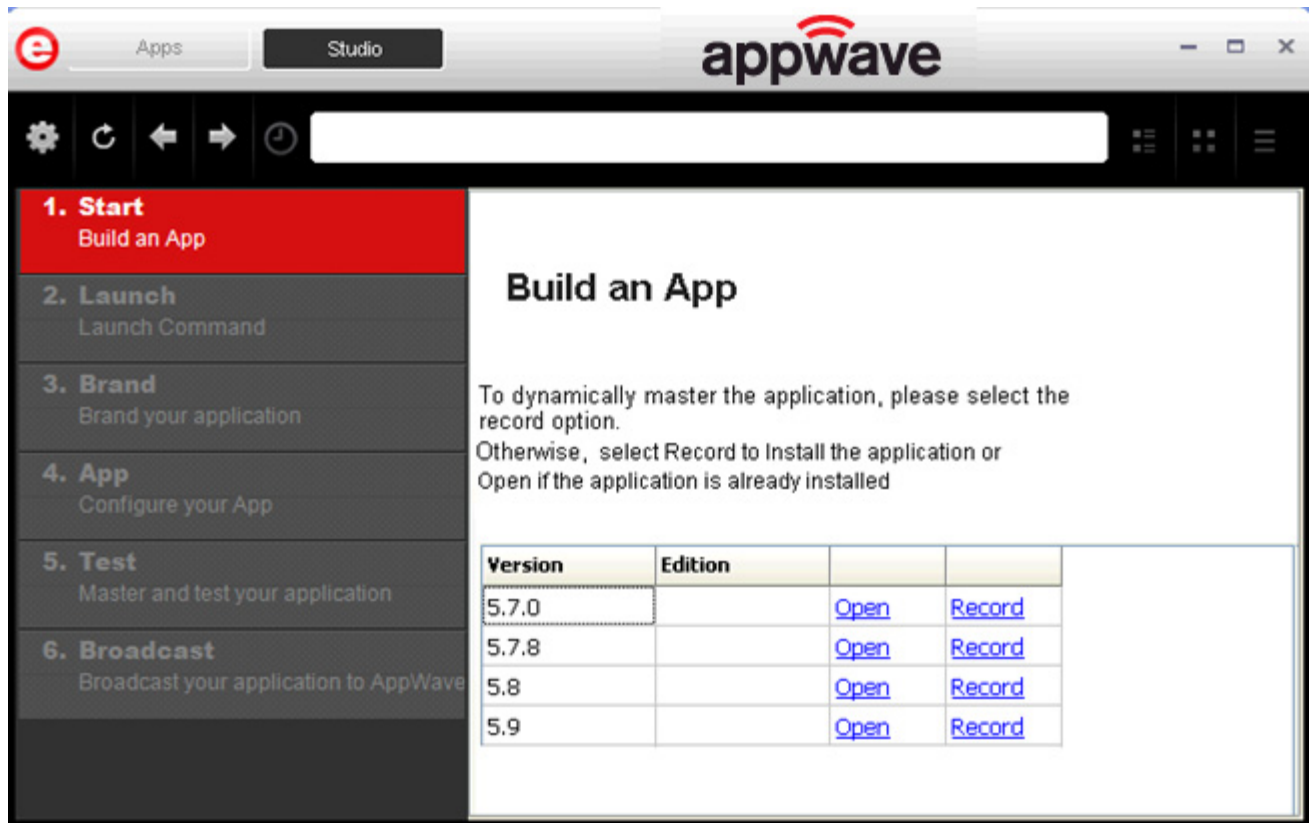
Open Option

1 Click **Open** if you want to open an installation that is already installed and the AppWave Studio dialog appears. This option opens a .tca file from the template which means that the app is already installed on your computer.
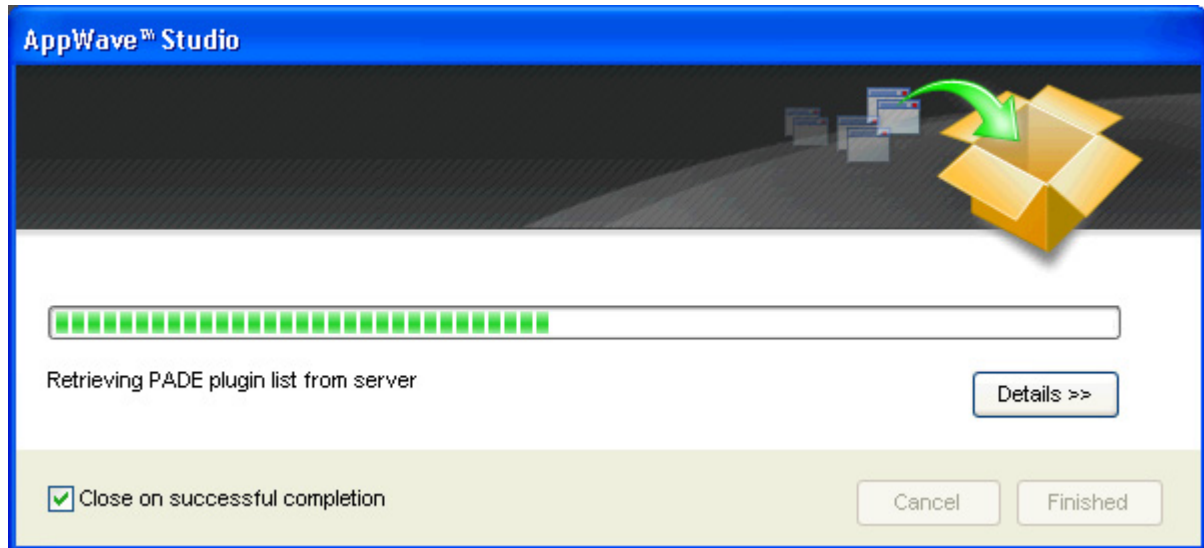
2 Once the retrieval process is completed click **Finished** and the app is launched on your computer.

Record Option

1 Click Record and the **Record <product name>** dialog appears.

   NOTE: Click the web site URL if you have not downloaded the executable for the selected app you are going to master. This executable needs to be saved in the `c:\`directory.

2 Enter the executable in the Installer Path field.
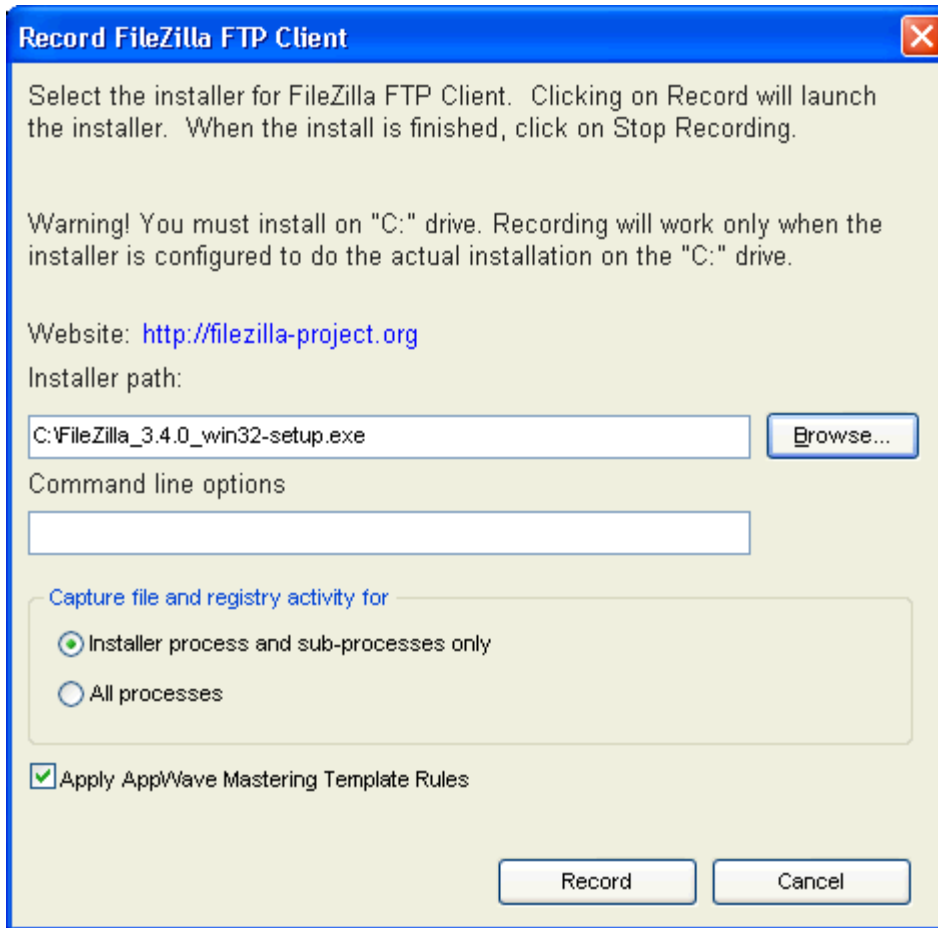
RECORDING

1   Once you have completed the information click **Record** and the install process starts.

2   A Recording dialog appears in the upper right corner of your screen and the AppWave Browser is minimized. The blinking recording status button indicates that recording is in progress.

3   Follow the install wizard for your selected app.

4   The licensing of your app depends on what type of AppWave licensing you have purchased as well as the type of app you are creating. For a full explanation of how to license your app, please refer to: <u>Managing Licensing</u>.

5   On the final panel of the Setup wizard, you may have the option to launch the app. Make your selection and click **Finish** to close the wizard.

> **TIP:**   You may not want to select the Launch option so you can investigate the registry keys that are created. If the roots of these keys are not in the app it is a good idea to include them so they will be managed.

6   Once the installation is complete, click **Finished** in the Recording dialog and the **Record Installation** dialog appears. This dialog identifies everything that needs to be installed. A progress bar shows each step of the process.

> **NOTE:**   Click **Details** to see the details of the installation while it is broadcasting or after it completes. If you want to see it after the installation is complete deselect the "Close on successful completion" option and then click **Details**.

7   Click **Finished** in the Record Installation dialog and one of several options are available depending on the template and executable in use.

   NOTE: At this point in the mastering process you are now working in the Studio portion of the AppWave Browser.

• The Launch page is where you supply the name of the launch executable. If this page appears you may need to supply information for the required fields such as the executable and the working folder.

OR

• The Brand page is where you provide branding information for the app. If this page appears you need to supply missing information such as Publisher, Display Name, or Product Name.

OR

• The Broadcast page is where you broadcast your mastered app to the AppWave. This page appears when the information is complete in the previous five steps and require no action on your part before broadcasting.

8   To complete this last step click **Broadcast**. You will get a dialog asking you to save the changes to your current working file. Click **Yes**, save the file in *C:\Documents and Settings\<username>\Embarcadero\AppWaveStudio\Output directory,* and the Broadcast dialog appears.

9   This dialog tracks the process of the broadcasting. Once you get the indication that the broadcast was successful, click **Finished** and the Broadcast dialog closes.

RUNNING THE APP

Once your app is broadcast, the newly mastered app build is placed in a working folder. The following screen shot shows one app has been placed in "KathyB's Folder" designated by the 🔴, as well as the app that was just mastered.

   NOTE: You can gain access to details and reviews for this App, as well as write reviews.

OPEN AN EXISTING PROJECT

When you created an App, all the project information is stored in a .tca file. You can revisit this project file and modify the project as needed. Each step in the project is then available for viewing and editing.

This option is only available through the AppWave Studio. A user must be granted authoring permission by an Administrator.

For more information see: Group Management.

1   Open the AppWave Browser and click Studio.

2   Click and select **Open Project**.
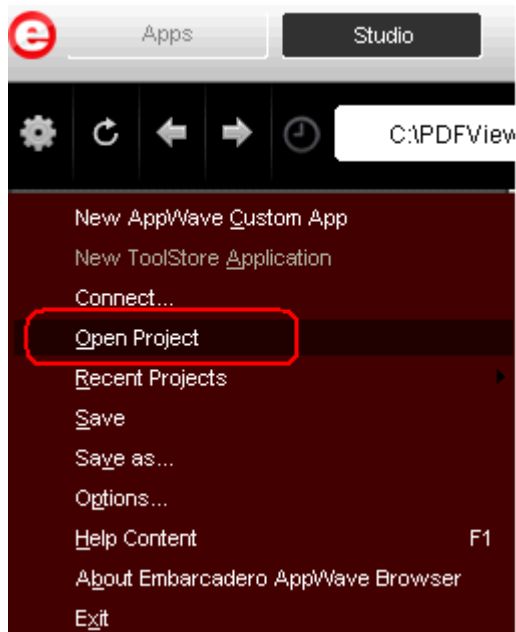
3   In the Open dialog, select a Studio file (.tca) and click **Open**. These files should be stored in *c:\Documents and Settings\<username>\Embarcadero\AppWaveStudio\Output.* The Open File dialog opens.
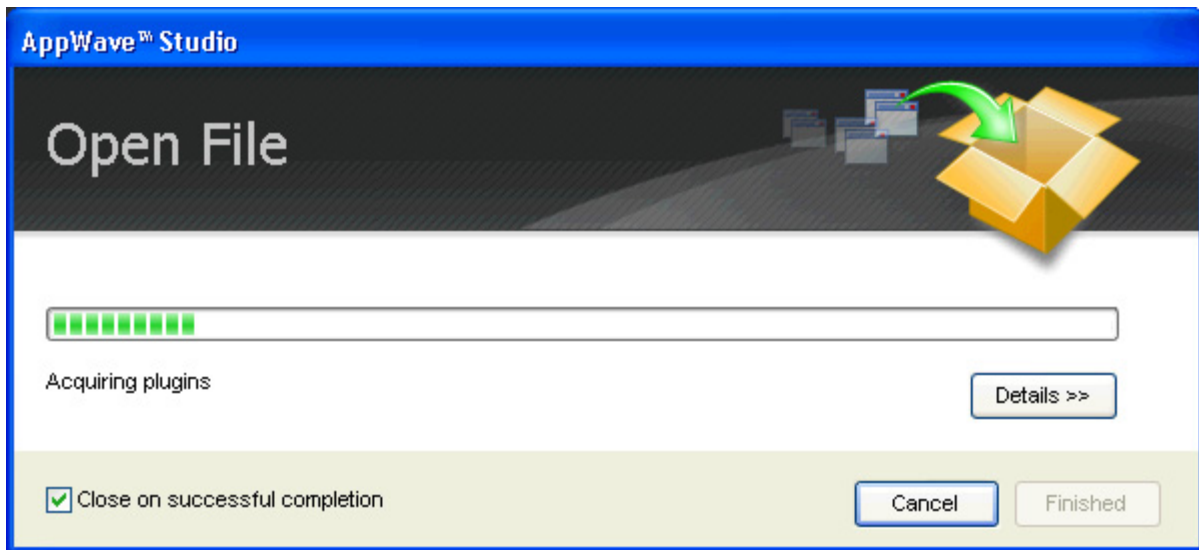
4   Once the file is loaded, the Broadcast page opens in Studio.

5   Click **Launch**. This is where you can edit existing values: the launch executable and working folder.
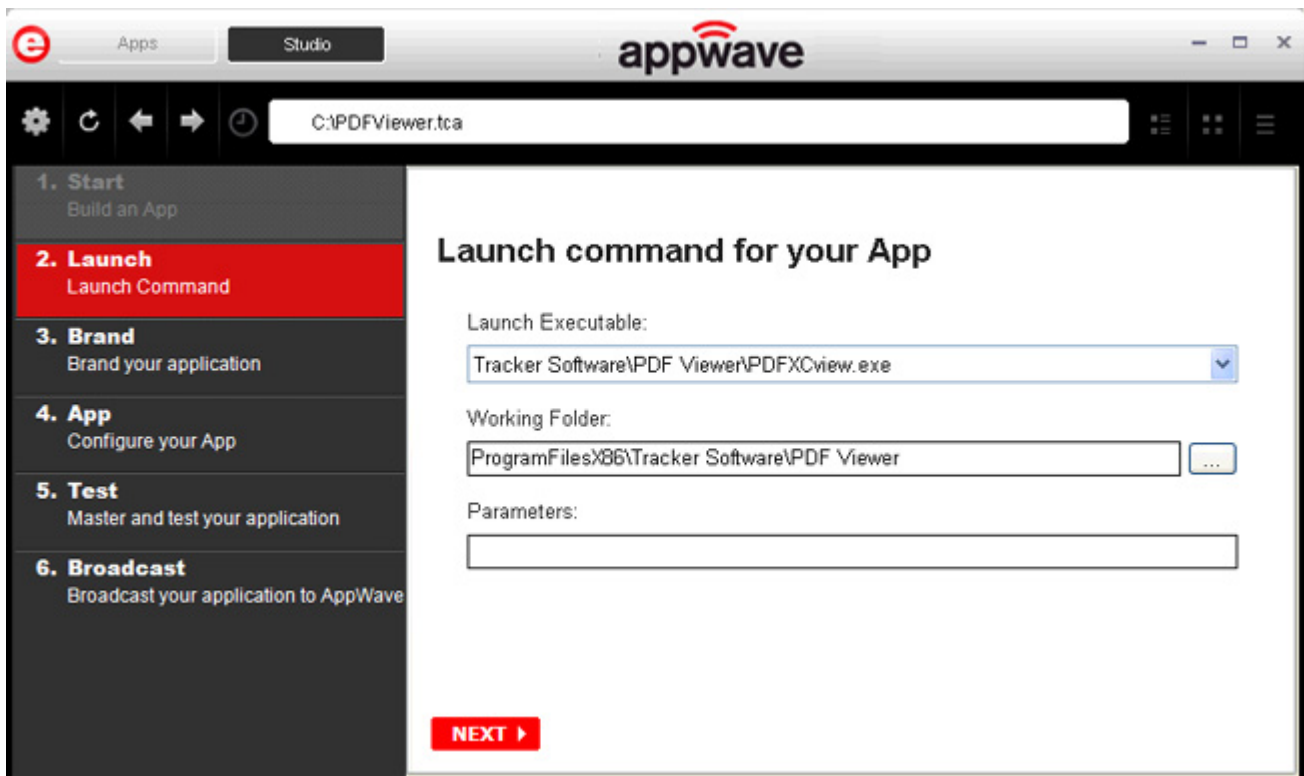
6   Click **Brand**. All the required fields are populated and you can edit existing values in all the fields.
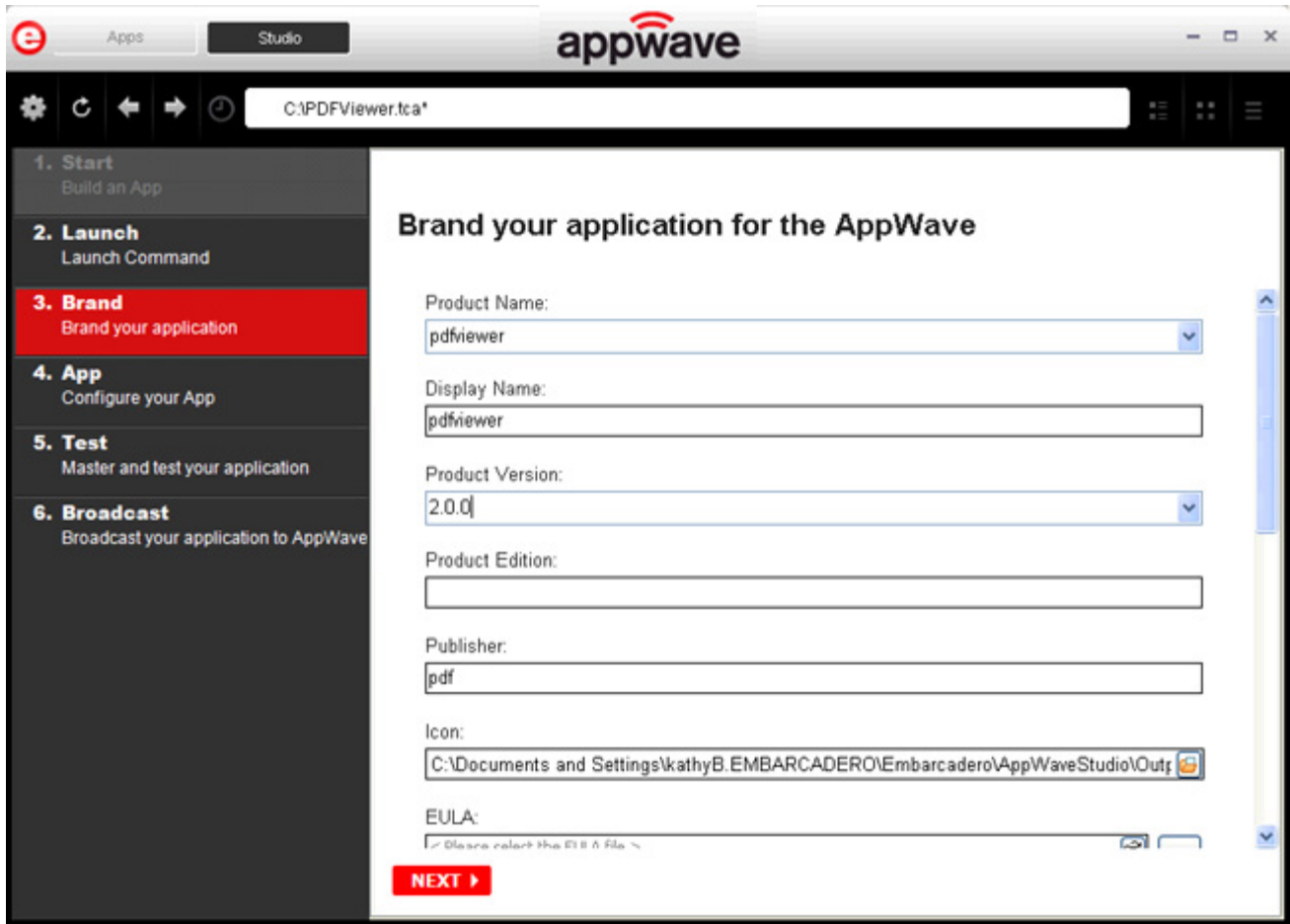
7   Click **App** and the five tabs are displayed: Overview, Files, Registry, Environment Variables, and Runtime).
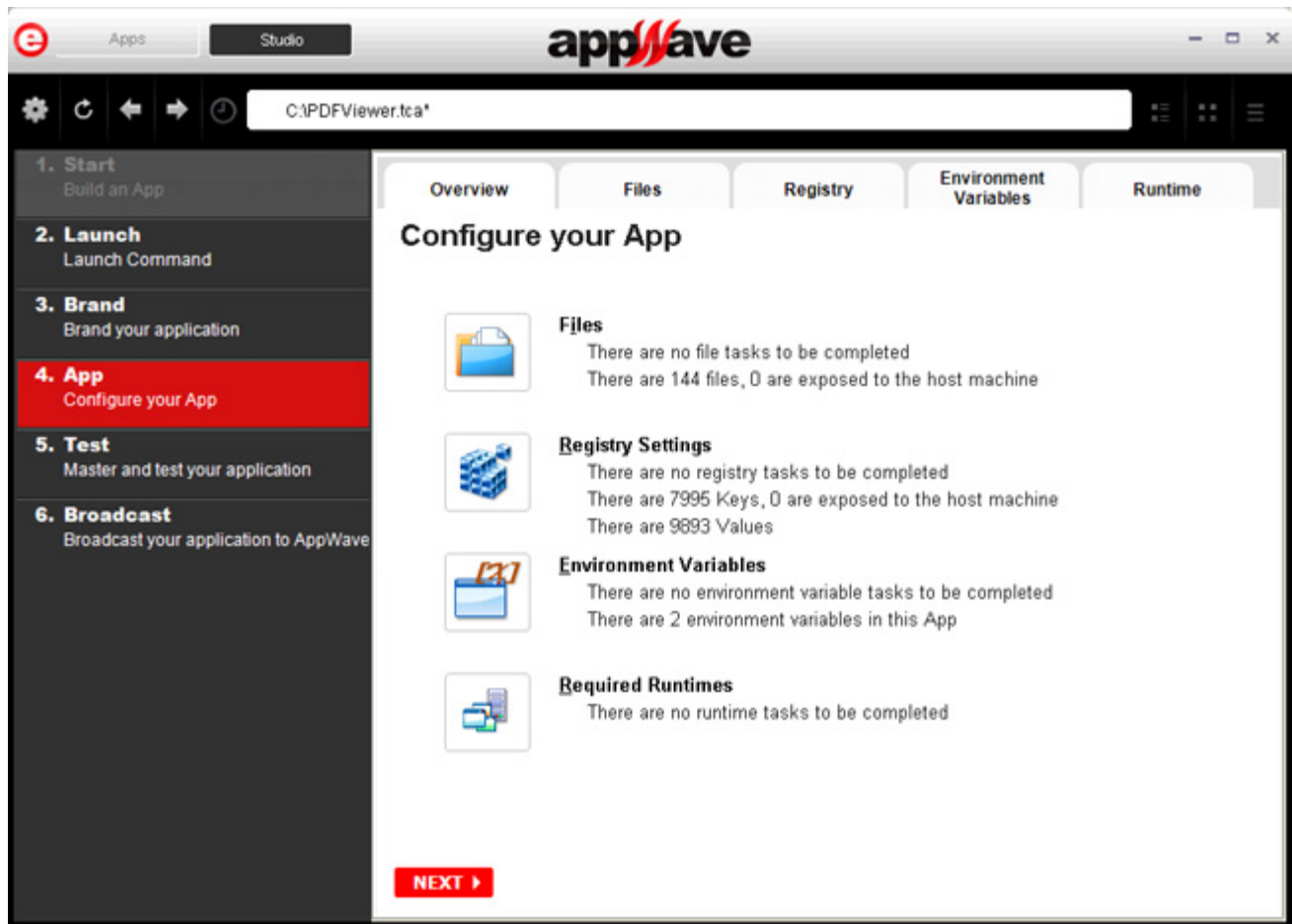
8   Click **Files**. You can add, remove and create Folders. You can also edit the properties of the folders and files.
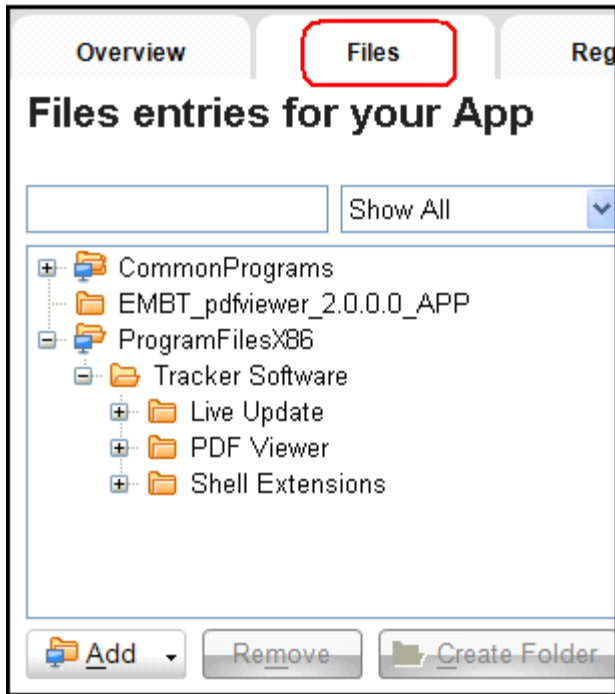
9   Click **Registry.** You can add, remove and create new registries here. You can also edit the
     properties of the registry keys.
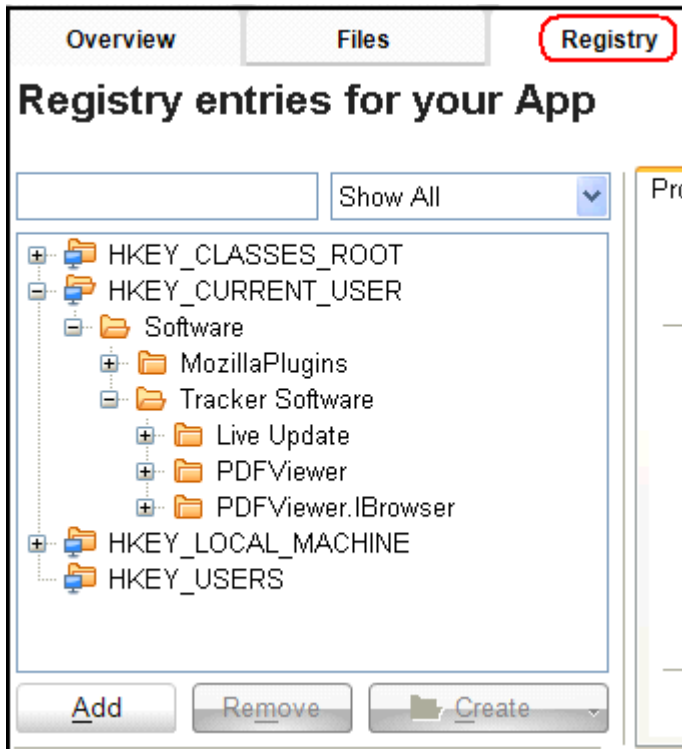
10   Click **Environment Variables**. You can add, edit, or remove these variables.
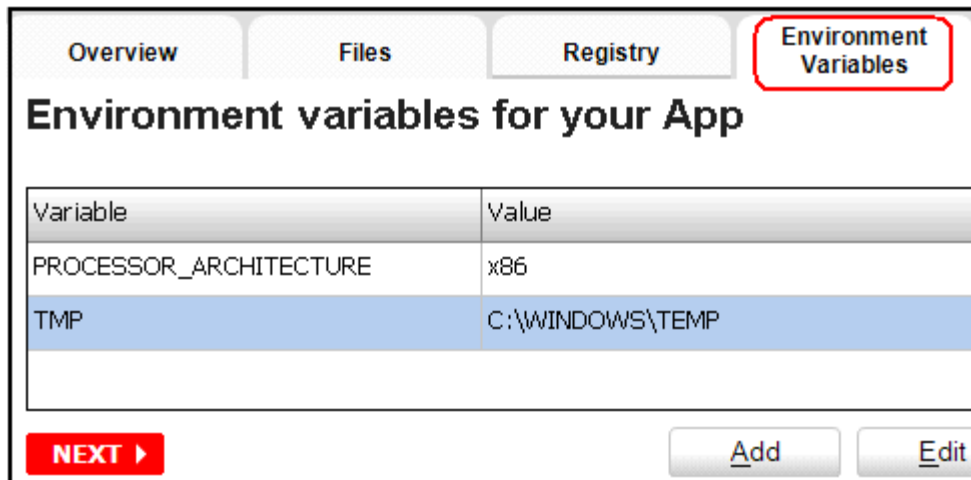
11 Click **Runtimes** and the runtimes required by your app are listed. You can select additional runtimes.
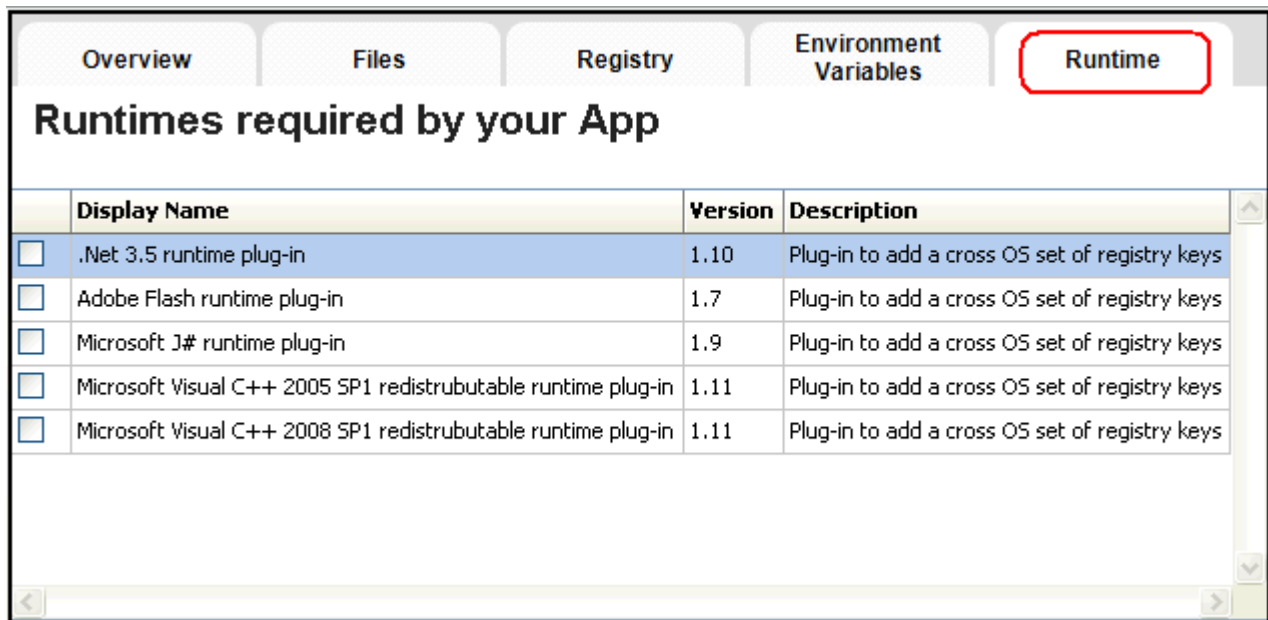
12 Click **Test.** You can edit existing values. If you made any changes in previous steps, for example if you added a runtime, you will be prompted to save the current working file.
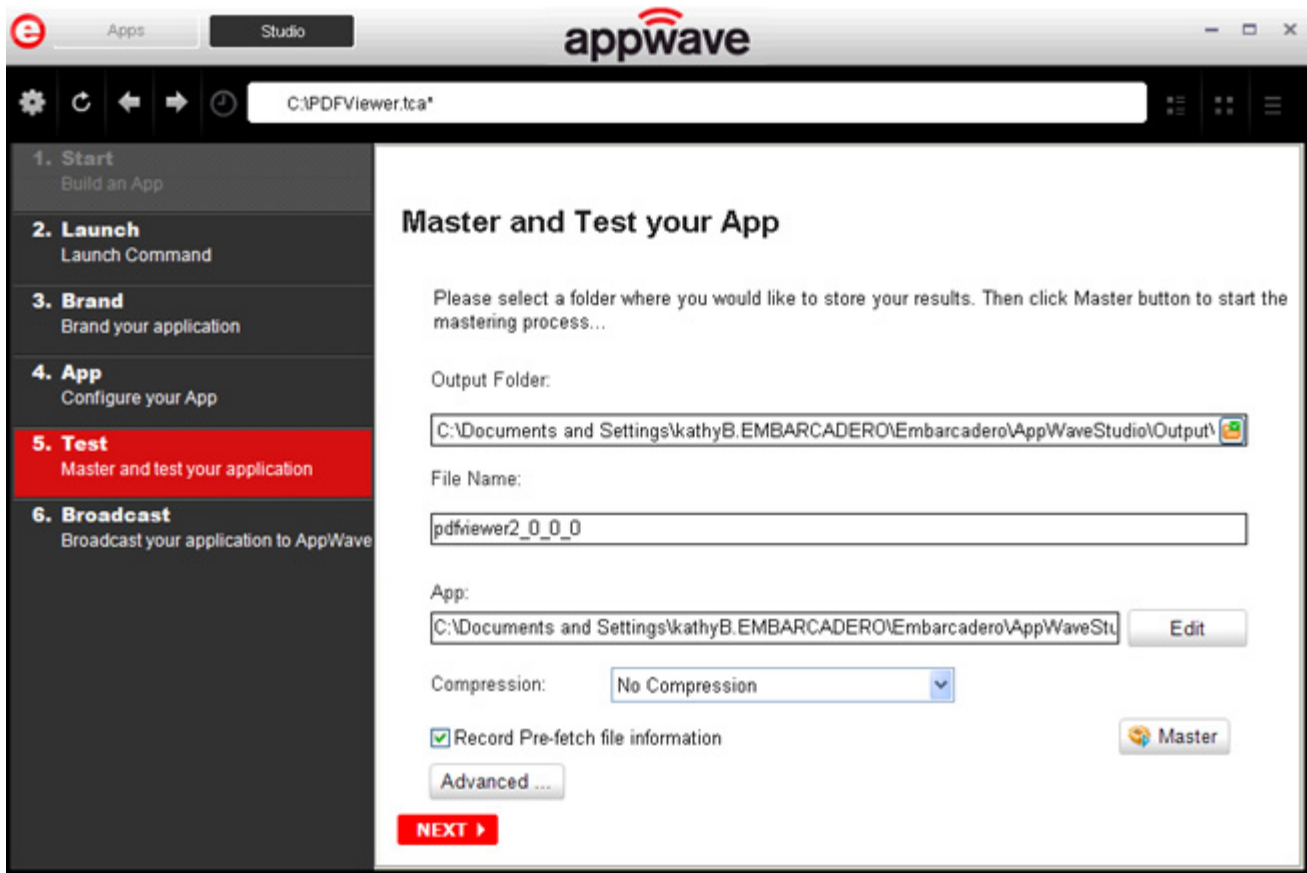
13 Click **Broadcast**: The "Always remaster projects before broadcasting to AppWave" option is
selected by default. Best practice is to always remaster a project before broadcasting.
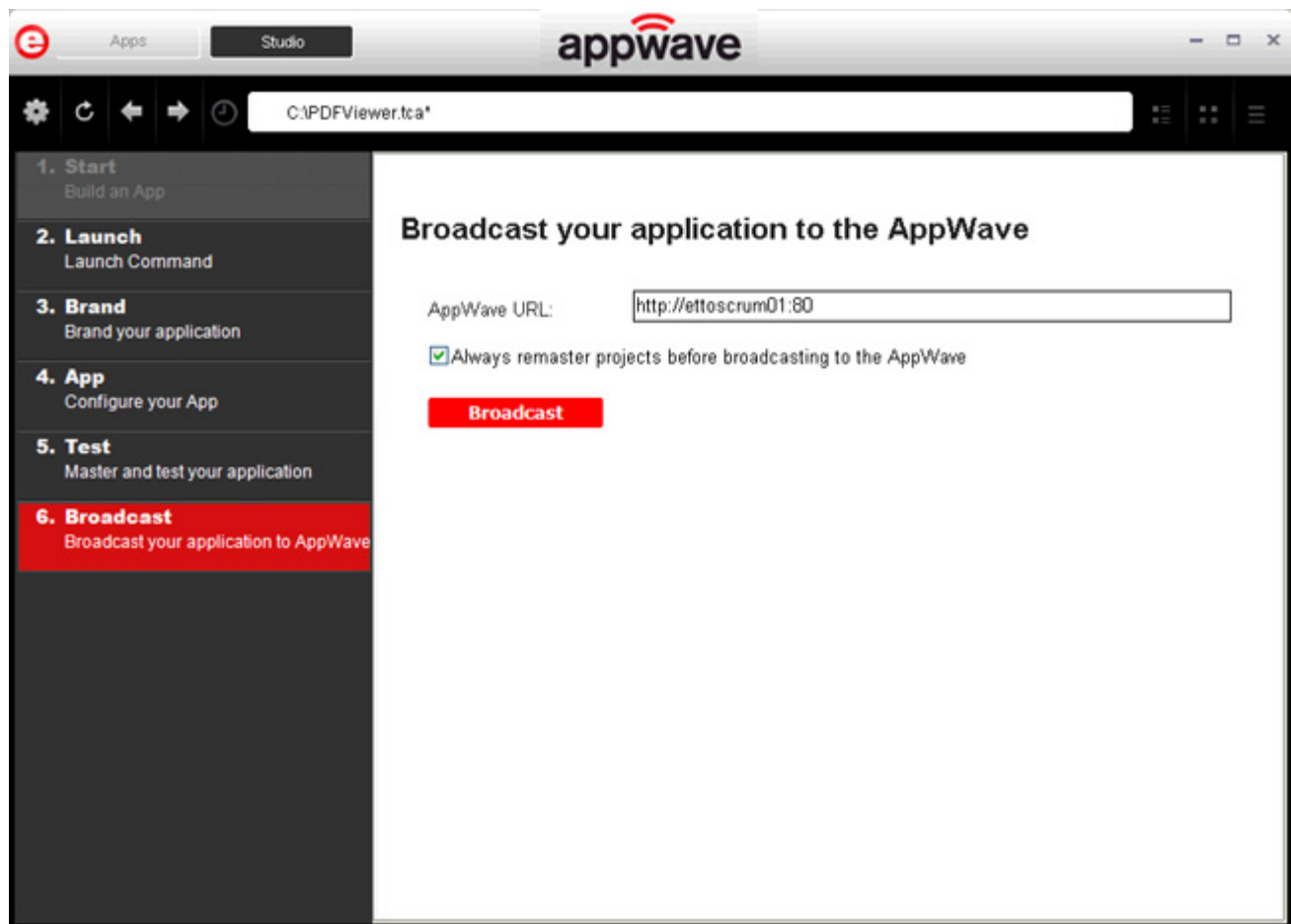
14   Click **Broadcast** and the revised app is made available in AppWave.

APPWAVE STUDIO ADMINISTRATOR

The Administrator plays a key role in the AppWave Studio and has the following functions and responsibilities:

Function	Description
Administrative Functions	Has access to all the administrative functions in AppWave. The Administrator can access the functionality through a web browser or through the Admin tab in the AppWave Browser.
Group Management	Has the option to give users access to the AppWave Studio so they can master applications. This mastering permission designates a user as an "Author" and gives the user the access.
Managing Licensing	Manages license inventory, assignments, and connected users. Also ensures the correct information is available for Licensing Third Party Applications.
Updating Apps	You can manage your applications so you always have the latest version.

ADMINISTRATIVE FUNCTIONS

This topic is a general overview of the Administrative Functions. For complete information on each function please refer to the *AppWave Administrator Guide*.

Function	Description
Manage Users and Groups	Users access the product repository using AppWave Browser, to cache, run and upgrade products. You control the levels of access that users have to products on an individual or group basis.
Manage Folders	Manage Folders allows you to restrict users and groups to a certain set of products available on AppWave.

Function	Description
Manage Apps	Applications from Embarcadero are stored in the AppWave repository; the set of products is called the product catalog. The product catalog helps you manage product caching and versions.
Manage Licenses	AppWave stores product licenses and shares them with users who cache the products using AppWave Browser.
System Management	Through the AppWave system management panes you can configure system properties such as what users can see when they connect, how applications are cached and streamed, proxy server details, along with streaming server details and backup licensing host configuration.
Reporting	Once you have the server set up and running you can gather statistics to help you manage users, groups, and licenses through the Reports.

GROUP MANAGEMENT

As Administrator you have the authority to assign users to groups. Before a user can gain access to the Studio application in the AppWave Browser they have to be assigned to the "authors" group.

To assign a user to the Authors group:

1 Open the AppWave Host, click Admin and login as administrator.

2 Click **Users** in the Manage Users and Groups folder.
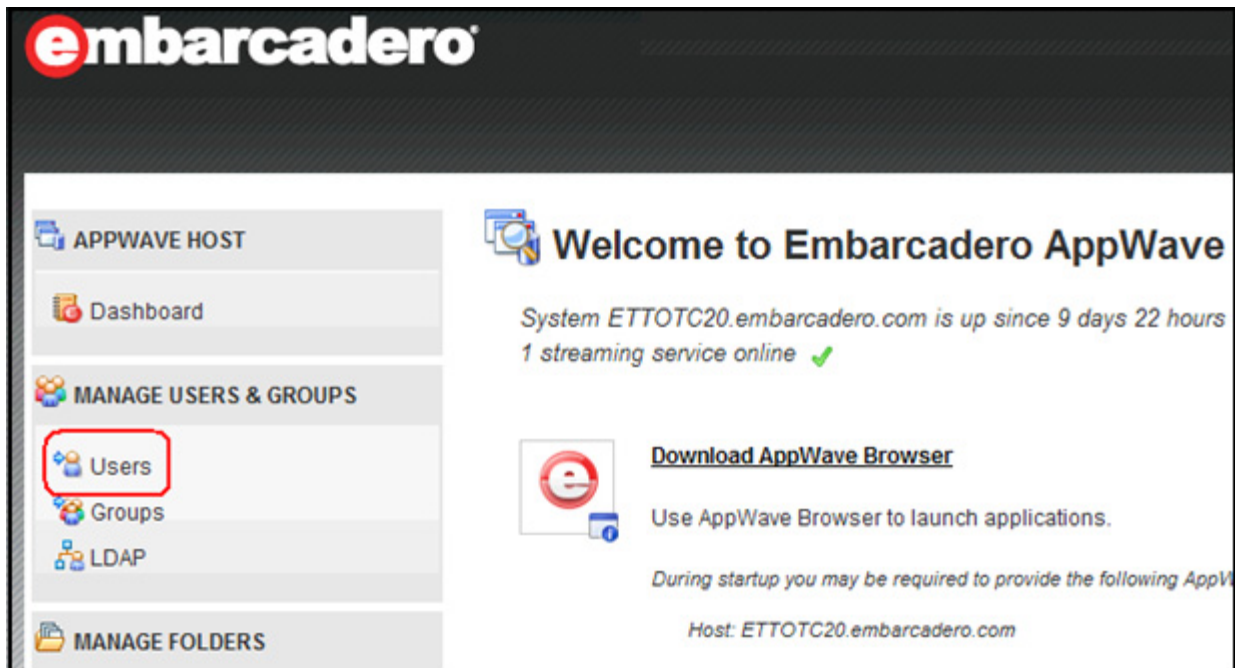
3   On the Users page that appears, click on your user name (or you can search for it as well). Your user page opens and shows your groups.

4   Click **Edit User** and on the User Details page that appears, click **Add Groups**.
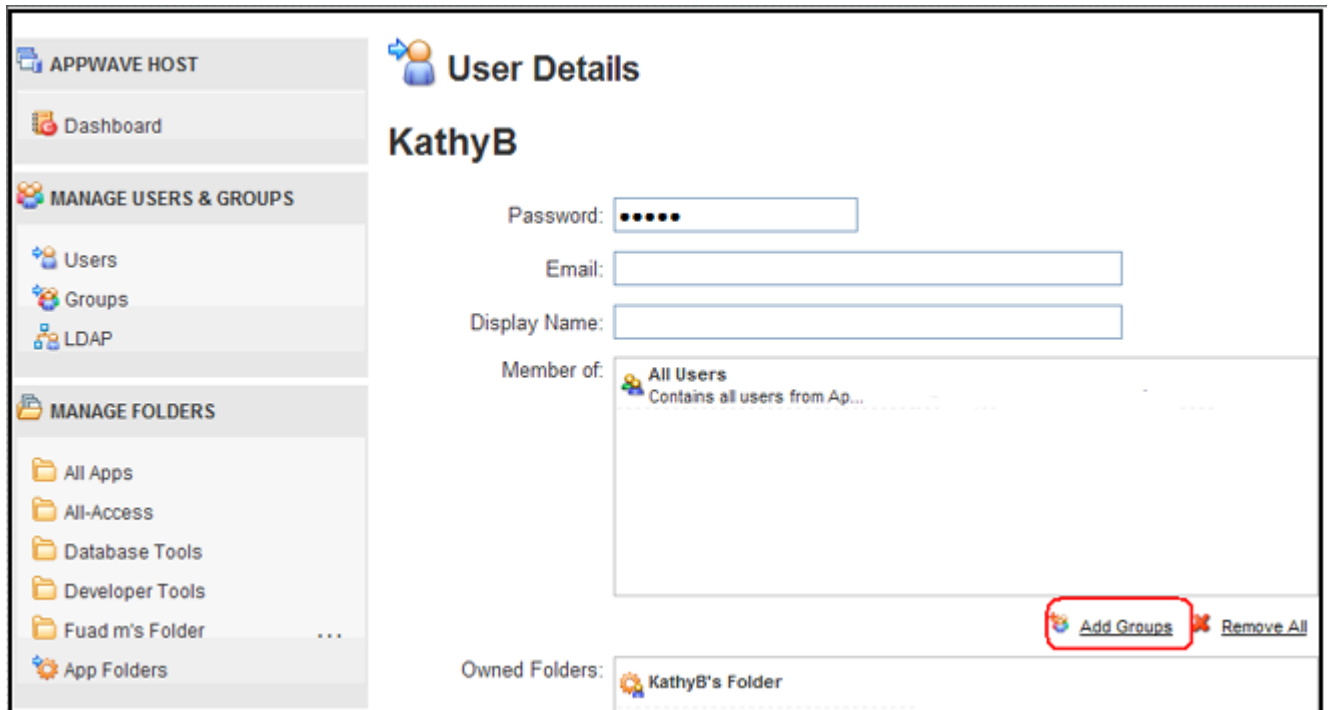
5   On the Groups page select **Authors** and click **Select**. The Groups dialog closes.
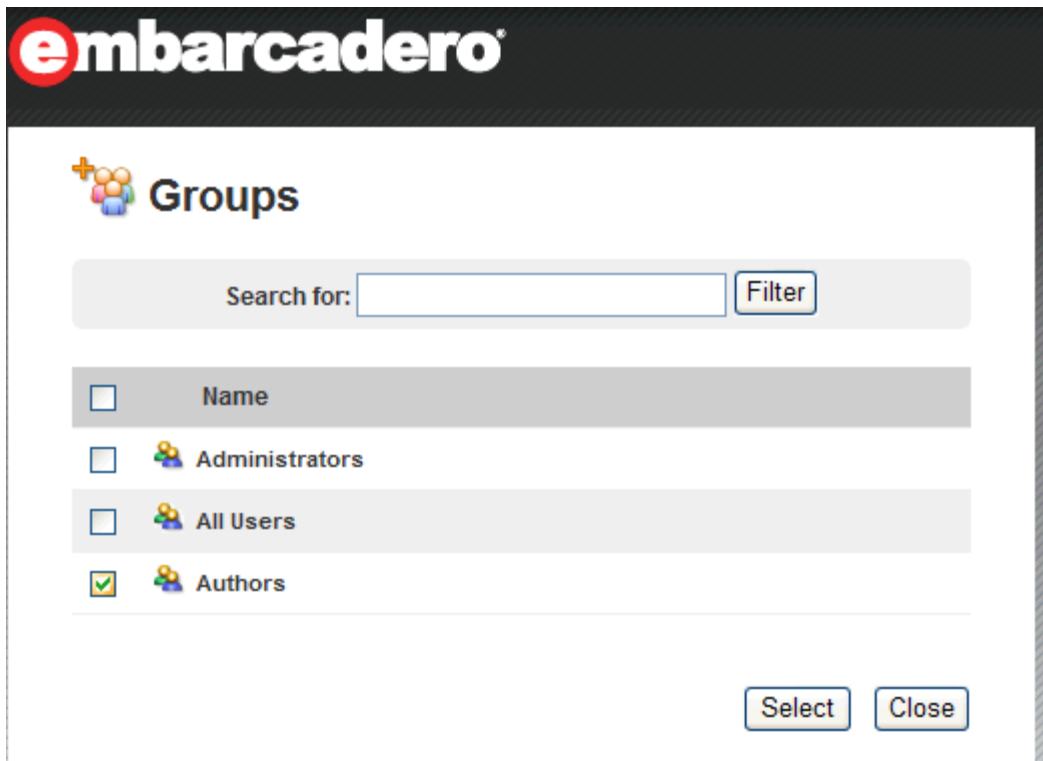
6   In the User Details page you are now listed as a member of the Authors group. Click **Save User** to save your changes.

For more information, please refer to the **Group Management** topic in the *AppWave Administrator Guide.*

MANAGING LICENSING

The Administrator manages licenses for the Apps that you master and use. If a license exists for the mastered app, it is automatically activated as you walk through the install process.

For more information, please refer to the **License Management** topic in the *AppWave Administrator Guide.*

AppWave includes Embarcadero products and supports the following types of Embarcadero licenses:

- **AppWave for Product Licenses (AW4X)**: These licenses are purchased by organizations from Embarcadero so they can master and run complicated commercial applications and run them in their own AppWave environment. The AW4X license has the following characteristics:

  - Concurrent and Network Named licenses are supported.

  - Administrators control the permissions and can deny a license for users/groups. Entries are saved in a userlist-concurrent.txt file (c:\Documents and Settings\All Users\Application Data\Embarcadero AppWave\elc\conf).

  - Administrators control reservations and can reserve x number of seats for users or groups. The number reserved must be equal or less than the total number of unit bought. The entries are saved in a groups .xml file (c:\Documents and Settings\All Users\Application Data\Embarcadero AppWave\elc\conf).

- **AppWave for Custom Apps License (AW4CA):** These licenses are purchased from Embarcadero to be used by customers to run applications they have mastered themselves without a template. Typically these are either their own organization's internal applications, or custom mastered commercial applications (not from a template). The AW4CA license has the following characteristics:

  - Concurrent licenses only.

  - Administrators control the permissions and can deny a license for users/groups. Entries are saved in a userlist-concurrent.txt file (c:\Documents and Settings\All Users\Application Data\Embarcadero AppWave\elc\conf).

  - Administrators control reservations and can reserve x number of seats for users or groups. The number reserved must be equal or less than the total number of unit bought. The entries are saved in a groups .xml file (c:\Documents and Settings\All Users\Application Data\Embarcadero AppWave\elc\conf).

- **AppWave Vendor Licensing (AW4V)**: These licenses are used by vendors to sell their mastered apps in the upcoming AppWave Store. A user purchases the app from the Store and can then run the app immediately. The AW4V license has the following characteristics:

  - Can be Named (workstation) or concurrent.

  - Administrators can deny a license for users or groups. This is based on product information specified in mytools.xml. The entries are saved in the userlist-concurrent.txt file (c:\Documents and Settings\All Users\Application Data\Embarcadero AppWave\elc\conf).

- **Trial Licenses**: A trial license allows a user to have full access to a product's functionality for a limited period of time, usually 30 days. In the License column of the AppWave Browser Apps Window, the license show as Trial. Users can launch or install the application as normal but will have only 30 days to use the application. The license then expires and the application is not available until a license is purchased.

ADDITIONAL INFORMATION

Any app mastered by a vendor will only ever accept a vendor license. If, however the customer masters a supplied vendor app, then that mastered app can use either a vendor license or an AppWave Product License.

- For free Apps where there is no charge for support, a vendor license can distributed. The owner (vendor) can then choose to take ownership of the app and the responsibility of maintaining it.

- If a customer has existing licenses from the vendor and wants to upgrade them into AppWave, this requires an AppWave Product License as well.

- A customer has the liberty to use a Vendor license like an AppWave Product License. However, the reverse is not true, because an AppWave Product License cannot grant entitlement to contained applications.

LICENSING THIRD PARTY APPLICATIONS

In order to launch third-party applications you need an AppWave for Custom Apps license. If a license already exists the AppWave host will have that license, which is assigned to the All Users group. For more information, see *AppWave Administrator Guide.*

License Automation provides a third-party product with a license when running that product using the AppWave Browser. The first part of implementing this licensing automation for third-party apps is to provide the license information in the AppWave Host.

1   Log onto AppWave.

2   Click **Manage Apps > App Inventory** to ensure the third-party application you want to license has been broadcast.

3   Click **Manage Licenses > License Automation** and all mastered third party applications that support license management are listed.

4   Locate the application to manage and in the **Usage** column, click the edit icon ( ) and the Licensing information for the application appears.

5   To add a new license or to change the information for an existing license, click **Add Activation Value**. To change the usage information for an existing license, in the Usage column for the relevant license, click the edit icon (✎)

NOTE: The fields that appear on this page depend on the application. It may be as simple as the total count of licenses and serial number; or it may ask for company ID, user ID, email information, etc.

6   Adding licenses:

- To add one license, in the **Total Count** field, complete the fields as required.

- To add more than one license click **Bulk Import**, point to the zip file containing the licensing slip files, and then click **Import**.

7   Click **Save** and the Auto Activation Licenses page now shows the number of licenses used and available. It is updated as licenses are used.

When the AppWave Browser user launches a licensed 3rd party application managed by AppWave, if there are licenses available the user a dialog opens providing them with a serial number for their use.

For more information on the various licenses, see [Managing Licensing](#).

UPDATING APPS

You can also ensure you always have the latest version of your applications by subscribing to AppWave notifications and by checking the Notifications folder in AppWave Browser, so you will always know when a new version is available.

For more information see: *Keeping your Apps Current* in the *AppWave Browser Apps User Guide.*

APPWAVE STUDIO OVERVIEW

The AppWave Studio application is integrated with the AppWave as a feature of the local library and appears in the AppWave Browser as the Studio tab.

The AppWave Studio is used to create an app file for an application. This application can use a mastering template or be created from scratch.

An application can be mastered into a file which can be run without installing it on your computer.This application runs on the desktop, locally but does not change the operating system or registry.It also does not share DLLs or other system-wide settings.The application data, including settings and licenses, are stored locally on the hard drive separate from the executable.The licenses are managed by the Administrator and are administered through Embarcadero's Sanctuary licensing system.

> **NOTE:** Once an app is mastered an "AppWave Template" is automatically created.This template file (.aw) contains all the information necessary to master an app such as production name, version, EULA, etc. It is placed in the output directory (c:\Documents and Settings\<user>\Embarcadero\AppWaveStudio\Output\).

There are three different ways to Create an App with One Click.Once your app has been mastered and then broadcast it can then be run on the local machine and be launched from either the local hard drive or the network using the AppWave.You can create and manage your tools in your Local Library and then share them with your organization or with the world.

For complete information about installing the AppWave Browser, see the *AppWave Administrator Guide.*

This chapter gives you an overview and explanation of all the AppWave Studio Elements.

- Windows Control Menu

- Toolbar

- Configuration Menu

- Connect

- Options

- Applications

- Tasks

- Steps

- Prefetch

- Mastering

- AppWave Studio Error Dialog

For information on how to use AppWave Studio, see: Using AppWave Studio.

APPWAVE STUDIO ELEMENTS

WINDOWS CONTROL MENU

Click the Embarcadero "E" icon ( ) to manipulate the window size of the AppWave Browser.

Command	Description
Restore	Restores the default window size.
Minimize	The AppWave Browser window is minimized and now appears on Window's task bar.
Maximize	Enlarges the AppWave Browser to full screen.
Close	Closes the AppWave Browser window and the application.

> **NOTE:** You can also double-click the window's title bar to maximize it or restore it to its previous size.

TOOLBAR

The following functions are available on the Studio toolbar:

Function	Icon	Description
Configuration Menu		Provides access to menu options such as connect, opening an existing project, or opening the online help.
Back		Once you have accessed a step and then moved forward, you can move back to that step.
Forward		Once you have accessed a step and then moved backward, you can move forward to that step.
Path	C:\PDF_XChange_Viewer.tca*	Displays the path, and the .tca file name of the application you are currently working with. It is also the search window.

> **NOTE:** The other items on the toolbar not listed in this table are used only by the Apps portion of the application.

CONFIGURATION MENU

Click the "gear" icon ( ) to gain access to the menu options. The following table explains each menu item:

Command	Description
New AppWave Custom	Takes you to the Build an App page where you can create an Apps from an Install executable or from scratch.
Connect	Opens the Connect dialog.
Open Project	Opens a dialog where you can browse to and select a .tca file (Studio Designer File) for an existing project.
Recent Projects	A pull-right menu displays the most recent projects you have worked on. Select a project and the Open File dialog appears. Once the file has loaded click Finished.
Save	Saves the .tca file for the project currently open in AppWave Studio.
Save As	Rename the current .tca file in the Save As dialog.
Options	Opens the General dialog where you can set system defaults.
Help Content	Opens the online help file for AppWave Studio. You can also press F1.
About Embarcadero AppWave Browser	Displays the current version and build number.
Exit	Closes the AppWave Browser application. **Note**: You do not get a warning dialog before the application closes.

CONNECT

When you click **Connect** in the menu the following dialog appears:

Options	Description
AppWave	This is the IP address that you want to connect to.The default setting is displayed. You can clear the history from this field.
Username	Your user name for the AppWave account you are using.
Windows Authentication	Your Windows credentials are used to log on. You need LDAP in order to connect to AppWave using Windows Authentication. Please refer to *Settings up the LDAP Connection in the AppWave Administrator Guide.*
Password	Your password for the AppWave account you are using.
Remember me	Select this option if you want the Browser to remember your password.
Login	The Loading Data symbol appears in the pane as you are connected.
Cancel	Cancels all selections and closes the dialog without taking any action.
Proxy Settings	Describes the proxy server used to connect to the Internet AppWave Browser.

Proxy Settings

When you click **Proxy Settings** in the **Connect to AppWave** dialog, the following selections are available:

Option	Description
Type	You have five different selections: **Do not use proxy**: A direct connection to the Internet is used. **Configure Manually Proxy Setting**s: You can specify Proxy server details. **Auto detect Network Proxy Settings**: When selected, the settings are automatically detected and you are connected to the Proxy server. **Use Proxy Configuration Script**: You can specify the Configuration Script that provides server configuration details. **Use Internet Explorer Proxy Settings**: Uses settings from the Internet Explorer.
Host	Displays the host name of the Proxy server.
Port	Displays the communication port used by the Proxy server.
Configuration Script	Enter configuration script needed to configure a Proxy server.
Bypass Proxy for Local Address	If you select this option, the proxy server is not used to connect to servers on your internet.
Bypass Proxy for Servers Starting With	Set a "wild-card" designation for all servers you want to bypass.
User Authentication	When selected, User authentication (user name and password) is required to log in.

OPTIONS

Click the "gear" icon ( ) and select **Options** to open the Options dialog.

Option	Description
Launch at Setup	Select this option to always have the AppWave Browser launch at startup.
Hide When Minimized	Select this option and when you minimize the AppWave Browser, the icon shows in the notification area but not on the task bar.

Option	Description
Cache Folder	This is where the log files are stored.The default folder is: c:\Documents and Settings\<user>\Embarcadero\Product Repository.You can browse to a new folder and select OK to set it as the default.
App Cache Size	The default cache size is 2048 MB.The maximum size shown is the size which is accepted by the machine.
	Click Clear to clear the cache.A dialog appears telling you when the cache is cleared successfully.
Turbo Option	This is a performance option allowing the AppWave Browser to run more efficiently. If the user doesn't have administrator rights at the time of install, the AppWave Browser installs without the Turbo Option.
	If the Turbo Option is enabled, when you run AppWave Browser, a file system driver is installed which allows AppWave to communicate with your hardware and operating system at a much lower level.
	The AppWave Browser and the AppWave applications you run while Turbo is enabled run faster because of the installation of the file system driver. By default, AppWave Browser runs in this mode if your Windows account has administrative privileges and if you are running AppWave Browser on a 32-bit machine.
	When Turbo is not enabled, AppWave Browser does not require Administrative privileges. This is a zero-install mode for locked-down desktops where your operating system user permissions do not allow you to install drivers. Performance is only slightly less than that provided in the Turbo mode.
My AppWave Name	Displays the name that appears on the Apps tab.The default is the user's name followed by 'Apps'.For example, 'Joe's Apps'.You can edit this name.
Restore Defaults	Restores all system default settings for the general options.
Save	Saves the changes to your General Options settings.

Option	Description
Cancel	Discards all changes made to the default settings and closes the Options dialog.

APPLICATIONS

Apps Tab: Opens the AppWave Browser application where you can view all groups and the products in each group. You can also view Notifications and Activity. For complete information on the Apps application, see the *AppWave Browser Apps User Guide*.
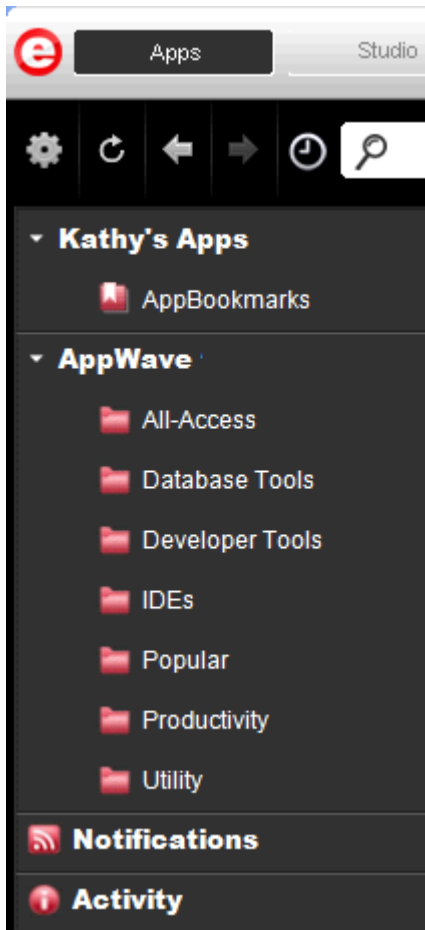
Studio Tab: Opens the AppWave Studio application where you can master Apps. The steps to create or modify an app are: Start, Launch, Brand, App, Test, and Broadcast.
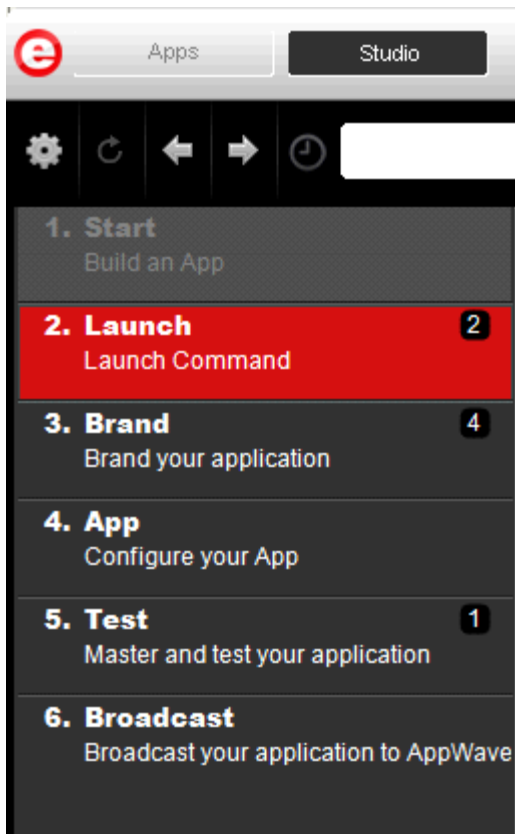
TASKS

The number icon located to the left of a step indicates the number of tasks that need to be completed before you can advance to the next step. In the screen shot below, Launch has two tasks to be completed.
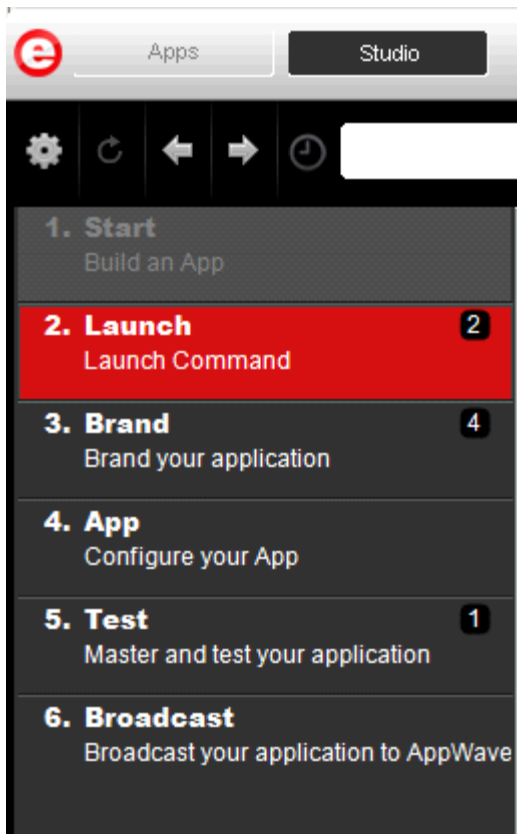
When you open the step, for example, click on Launch, a Tasks panel listing all the incomplete tasks appears at the bottom of the page. These tasks must be completed in order to proceed with the mastering. The fields requiring a "fix" have a warning icon (⚠). When you click **Fix** in the Task panel, you are taken to the appropriate task for the application you are configuring.
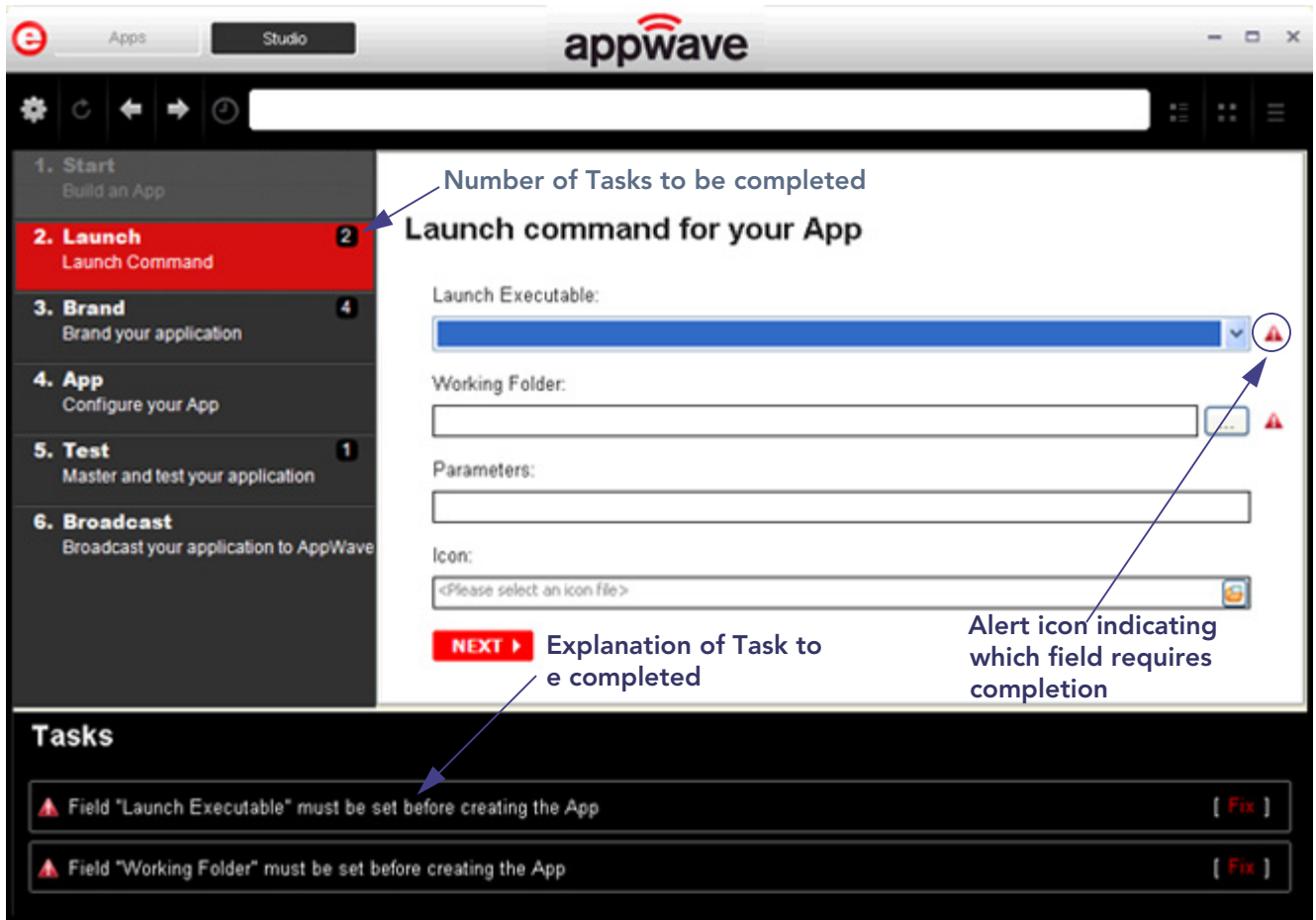
STEPS

The AppWave Studio uses six different steps to create and broadcast your tool. Each step contains task(s) that have to be performed in order to create an App. Many of these steps may be automatically completed during the creation and mastering of your App. The type of application you start with (i.e., from a template, an install, or from scratch) determines the level of detail you are required to complete.

The six steps in creating an app are defined below.

Label	Description
Using the Start Task	There are two options available to build a new App: create from install or create from scratch.Each has different operations that are necessary to complete.
	Note: Another option is to Create an App with One Click.This option is initiated in the Apps tab.

Label	Description
Using the Launch Task	The Launch command is where you select the launch executable.The working folder is then populated. You can also set any parameters needed for application and select an icon that you want to use.
Using the Brand Task	The product information entered here is used by the AppWave Host to identify it. This information includes name, version, edition, and publisher.It also includes web site information, system requirements, and product description.
Using the App Task	You can configure the following app elements: **Files**: Add, remove, or organize files and folders. **Registry Settings**: Add or create new registry entries. You can also view and change attributes. **Environment Variables**: Add and/or edit environment variables. **Required Runtimes**: When a product (for example the .NET framework required by RADStudio) needs an execution runtime, it is provided to those who master applications. Once you make a selection you are requested to save your changes to the current working file in order to analyze required runtime changes.
Using the Test Task	Opens the **Master and Test Your Application** page.This displays the output folder and file name. You can change the output folder and edit the file name. Note: Only ASCII characters should be used in File Name field. The mastered application will not launch if non-ASCII characters are used (for example Synérgie). When you click Edit in the App field, a dialog opens asking you to select the app file (extension .stp) to associate with the .tca file. You also choose the type of Compression you want to use and click the Advanced button to select an encryption type. When you click Master, a confirm dialog appears asking you to save the changes to your current working file. Click Yes and a Save As dialog appears.The file you save has a .tca file extension. For more information see: Understanding the Studio File. Once the save process is complete, the Mastering App dialog appears with a progress bar. Click Details to view the mastering details. Click Finished to close the dialog.

Label	Description
Using the Broadcast Task	This step requires that you select an AppWave URL. You can choose to remaster the project before broadcasting.This option is selected by default.

USING THE START TASK

When you click **Studio** the first step (Start) is active.This is where you select which type of app you want to create: from Install or from Scratch.

Create from Install

When you click **Create AppWave Supported or Custom App from Install** the Record Application Installation dialog appears.
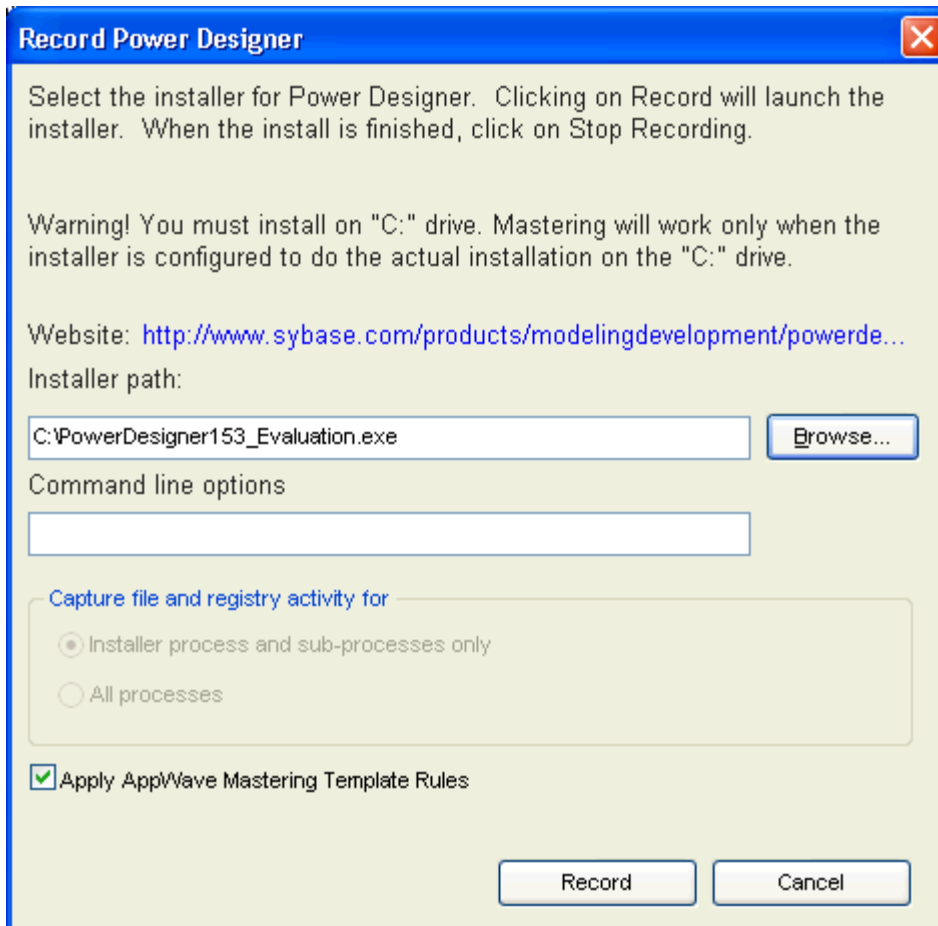
The dialog has the following fields:

Function	Description
Installer Path	Browse to the path were the installer of the application you wish to capture resides. You must install on the "C" drive or the recording doesn't work.
Command Line	The "Command Line" value indicates how to launch the file and the value is the trigger to run a specific command. For information on commands you can use, see Using the Command Line.
Capture file and registry activity for: • Installer process and sub-processes only • All Processes	Records only the install process. Records all the processes associated with the application.
Apply AppWave Mastering Template Rules	As the process goes through the steps to create an App, tasks may be created indicating missing data.This option automatically competes any missing data.This is the default setting.
Record	Click **Record** to launch the installer.The Recording dialog appears and the red light blinks indicating that the recording is in process. You can start recording, stop recording, retrieve recording log messages (if any), and configure what process to record (all or just the installer). You have two options in the Recording dialog: cancel the recording or finish the recording.
Cancel	Cancels the recording process and closes the Record Application Installation dialog.

Create from Scratch

When you click **Create AppWave Supported or Custom App from Scratch** the **Select Folders and Files to Include** dialog appears.
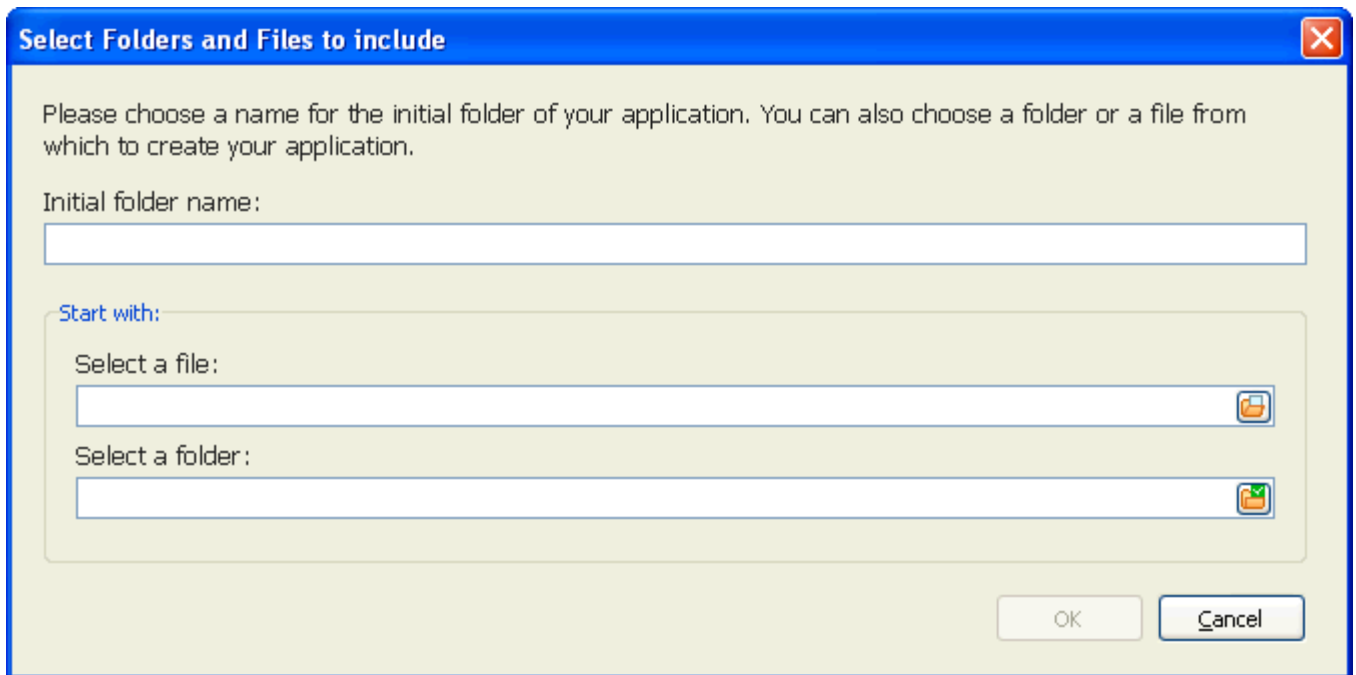
This dialog has the following fields:

Field	Description
Initial Folder Name	Create a name for the initial folder. This folder is the "working" folder for your application.
Select a file	Browse to the Installer path and select the executable.The Installer path must on the "C:" drive.For example: `C:\Program Files\Embarcadero\`.
Select a folder	Browse to an existing folder or make a new folder.This contains the executable from the "Create from Scratch" process.

USING THE LAUNCH TASK

The Launch page opens automatically once you click **Finished** in the Record Installation dialog. The following fields need to be completed before advancing to the next step.

Function	Description
Launch Executable	Select the executable from the drop-down list.This is a required field.
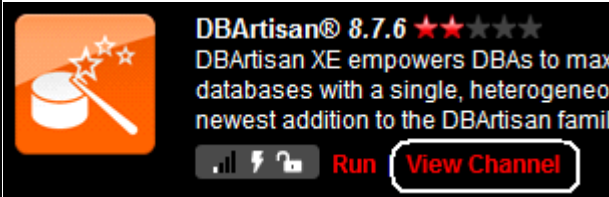
Function	Description
Working Folder	The working folder field is automatically populated once you select the executable.This is a required field.You can also browse to a different folder.
Parameters	Enter any parameters that are required by the launch executable.

USING THE BRAND TASK

Click **Brand** to identify the application for the AppWave. The Product Name and Product Version information entered here is used in the AppWave to identify it.

> NOTE: When you first open this page, any empty fields are listed in the lower Tasks panel. Click **Fix** on an issue and you are taken to the step and field where the fix is needed.

Field	Description
Product Name	Select the Product name from the drop-down list.
Display Name	Enter the name of the product. Select the name from the drop down list. This is a required field.
Product Version	Select the appropriate production version from the drop-down list.
Product Edition	Enter edition information so end user knows exactly which edition was used in the creation of the App.
Publisher	Enter the publish of the product. For Embarcadero products use *Embarcadero Technologies*.This is a required field.
Icon	Select the icon file (.ico) by clicking the drop-down icon and opening the appropriate folder.

Field	Description
EULA	End User License Agreement. Click the drop-down arrow in the EULA field and select the text file containing the licensing agreement information from the installed application's folder. The appearance of the EULA during the install depends on the type of license required for the application you are managing. If a EULA is required, the license agreement appears during the install process and the user must accept this agreement before the license verification process can proceed.
Screen Shot	Select a screen shot image that you want to appear when a user selects an App. The image as well as the details of the app opens when you click **View Channel**. The file format is jpg.
Web Site	Enter the URL for the application's web site.
Support Contacts	Enter the contact information for the application's company. This information is then available on the Apps tab for the installed application.
System Requirements	Enter any system requirements that are needed for the application to run.
Product Description	Enter any descriptive information. This information appears with the application in the list of application in the AppWave. This information is then available on the Apps tab for the installed application.
Long Product Description	Enter additional descriptive information here.This information appears with the application in the list of application in the AppWave.
Release Date	This date is when the application was released to the public for general use.

USING THE APP TASK

You can configure your app by viewing what was created when you built a new App. You have access to the following:

Icon	Name	Description
	Files	Click Files to display the file entries for your app.
		This lists all the files that are included in your app. It also displays the properties of each file.The information is displayed in either a list or grid format.
		You can add or remove files and folders, or create a new folder.
	Registry Settings	Displays any existing registry keys that were created when the install file was read.
		You can add new registry entries and then view and change the attributes of that new entry.
	Environment Variables	Lists the environment variables for your app. You can add new variables, edit existing variables, or remove variables.
	Required Runtimes	When a product (for example the .NET framework required by RADStudio) needs an execution runtime, it is provided to those who master applications. Once you make a selection you are requested to save your changes to the current working file in order to analyze required runtime changes.

FILES

To gain access to this dialog click the App step and the **Configure your App** dialog appears.

Click **Files** or [icon] to open the **File entries for your App** page. The left-hand panel displays all your files.You can add or remove files. You can also create a new folder under a selected folder.

You have a drop-down box where you can select the Show All (visible only to the App) or Exposed to Host (visible on the host machine).

The Properties and Grid tabs to the right displays information on a selected file.
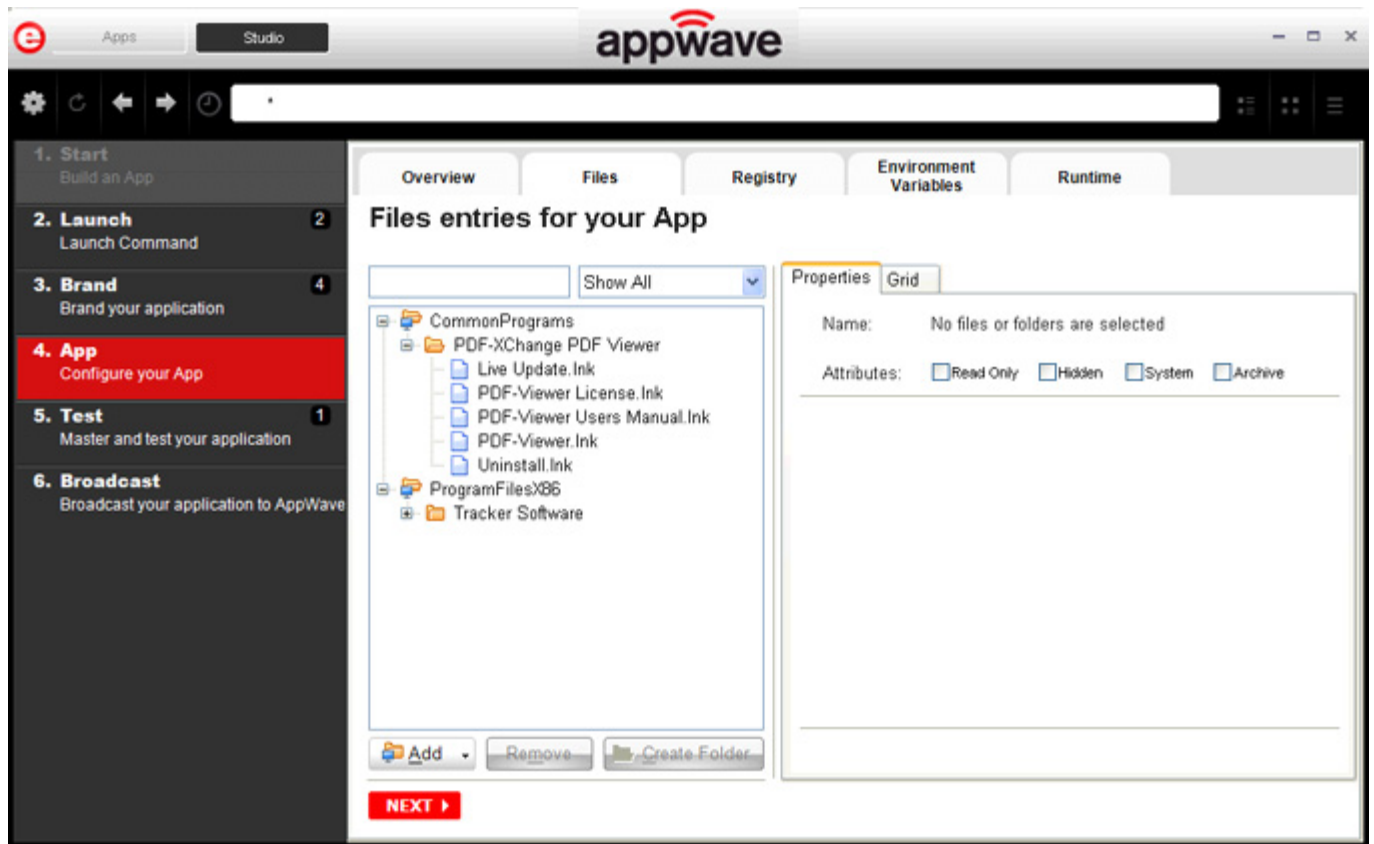
File Actions

You can add or remove files and folders, or create a new folder. There are addition commands available under **Add**.

Button	Description
Add	When you click the drop-down arrow you have three options.
	File: The Open dialog appears.Browse to the file you want to add and click **OK**.
	Folder: Opens the Browse for Folder dialog. Select the folder you want to add.
	Virtual Folder: Opens the Add a Predefined Virtual Folder dialog. Select a folder from the drop-down list and click OK.
	Virtual folders are not dependent on location in a hierarchical directory. So a Streaming core allows you to basically mount the file system so that when you want to add to your App, and remaster it, you have the virtual file system present on the hard drive so you can do whatever you want with it.
Remove	Select a file in the tree and click **Remove**. A Confirm dialog appears asking you to click **OK** to continue.
Create Folder	Select a folder in the tree and click **Create Folder**. Enter a name for the new folder and it is placed as a subfolder in the tree.

You also have commands available when you right click on a folder or file in the tree.

Command	Description
Add File	The Open dialog appears. Browse to the file you want to add and click **OK**.
Add Folder	Opens the Browse for Folder dialog. Select the folder you want to add.
Add Virtual Folder	Opens the **Add a Predefined Virtual Folder** dialog. Select a folder from the drop-down list and click OK.
Create Folder	Select a folder in the tree and click **Create Folder**. Enter a name for the new folder and it is placed as a subfolder in the tree.
Remove	Right-click a file in the tree and click **Remove**. A Confirm dialog appears asking you to click OK to continue.
Apply Visibility to Children	Takes the parent folder visibility settings and applies them to all the children.

File Filters

You have a drop-down box where you can select the **Show All** (visible only to the App) or **Exposed to Host** (visible on the host machine).

Properties/Grid Tabs

Information on a selected file/folder is displayed in two different formats: properties and grid.

The properties tab for a file displays the following information:

Option	Description
Name	If you selected a file, the file name and path of the selected file.If a folder is selected, the number of files and/or folders is displayed.
File Size	If application, the size of the file is displayed.
Attributes	The attributes of a selected file are displayed: Read Only; Hidden, System, Archive (default setting)
What applications have access to this file	
Only the App	The selected file or folder is only accessible within the app and is not visible to the host machine. Files or folders with attribute changes are kept private in a cache managed by the Browser.
All applications	The selected file or folder is available to the host machine and can be accessed by external processes and users. Unless it is hidden from the folder listings, it is displayed by any and all applications that list the contents of the host machine's file system.
In the case of a conflict, which file should be used?	
Local System Version	Before activation, the app checks to see if this file already exists on the host machine. If it does, it uses the existing file.If the file does not exist, it is permanently placed on the machine.
App versions	If this file already exists on the host machine, the app version of the file replaces the existing file as long as the app is active in the Browser. If you remove the app or clear the local Browser cache, the previous version is restored.
Security Settings	

Option	Description
Hide From Folder Listings	This file is not displayed in the list of folders. Select this option if you want to hide this particular file in a file listing so it is not visible to other users.
Prevent Copying	Select this option if you do not want other users to copy the file from a temporary location where it was streamed. However these files are able to be read from outside.
Make Read Only	Select this option if you do not want other users to alter the files.
Do not cache in the local machine	This file is not permitted to be cached to the local system. This security setting ensures that the file is not located anywhere on the system.

The properties tab for a folder displays the following information:

Option	Description
Name	If you selected a file, the file name and path of the selected file. If a folder is selected, the number of files and/or folders is displayed.
Contains	
What applications have access to this file	
Only the App	The selected file or folder is only accessible within the app and is not visible to the host machine. Files or folders with attribute changes are kept private in a cache managed by the Browser.
All application	The selected file or folder is available to the host machine and can be accessed by external processes and users. Unless it is hidden from the folder listings, it is displayed by any and all applications that list the contents of the host machine's file system.
Should the App version of the item permanently replace the version on the Host?	
Permanently Replace Host Version	Before activation, the app checks to see if this file already exists on the host machine. If it does, it uses the existing file. If the file does not exist, it is permanently placed on the machine.

Option	Description
Temporarily Replace Host Version (Turbo mode only)	If this file already exists on the host machine, the app version of the file replaces the existing file as long as the app is active in the Browser. If you remove the app or clear the local cache, the previous version is restored.
Security Settings	
Hide From Folder Listings	If this file is visible on the host file system, it is not shown to users and processes requesting directory listings from the operating system, however it is available if explicitly accessed by name.
Make Read Only	The Browser does not allow this file to be changed when broadcast with the App.
Prevent Copying	If this file is visible on the host file system AppWave blocks other processes from copying it.
Do not cache in the local machine	This file is downloaded from the AppWave or a Streaming Server whenever it is first used by the App. It is not stored locally between running instances of the product.

The grid format displays the following for both the file and folder:

Option	Description
File/Folder Name	The name of the file or folder selected.
Size	If a file, the size of the file.If a folder, then the number of files/folders contained in the folder.
Application Access	The selected file or folder is available to the host machine and can be accessed by external processes and users. Unless it is hidden from the folder listings, it is displayed by any and all applications that list the contents of the host machine's file system.
App	The selected file or folder is only accessible within the app and is not visible to the host machine. Files or folders with attribute changes are kept private in a cache managed by the Browser.
Path	Path where the file/folder is stored.

To gain access to the attribute information in the grid format, select a file in the grid and click **Edit**. The Edit selection dialog lists all the attributes for the selected file.

You can delete a file by selecting the file and clicking **Remove**. A confirm dialog appears.

REGISTRY SETTINGS

When you click Registry Settings or the icon ( ), the **Registry Entries for your App** dialog appears. You can:

- Add a registry key from the system registry

- Remove a registry key you have added

- Create a registry key or value

On the left is a tree of registry setting groups. The base folders are defined in the .tca model:

HKEY_CLASSES_ROOT

HKEY_CURRENT_USER

HKEY_LOCAL_MACHINE

HKEY_USERS

You have a drop-down box where you can select the Show All (visible only to the App) or Exposed to Host (visible on the host machine).

CAUTION: It is best practice to review the registry settings before broadcasting your application. Make sure there are no files or registry keys not directly relating to the application. Remove any that have been included in error.

You need to also verify that registry keys refer to a generic path. For example a path should be *%USERPROFILES%\My Documents* not *C:\Documents and Settings\<username>\My Documents*.

View and Change Attributes

In the right-hand panel you can view and/or change the attributes of a selected registry entry.

Command	Description
What applications have access to this registry	
Only the App	These are items accessible only within the App, and not on the Host. The selected file or folder is only accessible within the app and is not visible to the host machine. Files or folders with attribute changes are kept private in a cache managed by the Browser.

Command	Description
All Applications	The selected file or folder is available to the host machine and can be accessed by external processes and users. Unless it is hidden from the folder listings, it is displayed by any and all applications that list the contents of the host machine's file system. **Note**: See caution below when mastering an app on the Windows 7 OS.
Should the App version of the item permanently replace the version on the Host?	
Permanently Replace Host Version	Before activation, the app checks to see if this registry already exists on the host machine. If it does, it uses the existing file.If the file does not exist, it is permanently placed on the machine. **Note**: See caution below when mastering an app on the Windows 7 OS.
Temporarily Replace Host Version (Turbo mode only)	If this registry already exists on the host machine, the app version of the file replaces the existing file as long as the app is active in the Browser. If you remove the app or clear the local Browser cache, the previous version is restored.

CAUTION: When mastering an app on the Windows 7 OS, create registry keys under HKEY_CLASSES_ROOT or in one of the subdirectories of HKEY_CURRENT_USER, HKEY_LOCAL_MACHINE, and HKEY_USERS.

If the properties for the key you created are set to "All Applications" and "Permanently Replace Host Version" an error message (APM_E_APP_CONFIG_INSTALL_REGISTRY) will be generated.

ENVIRONMENT VARIABLES

When you click Environment Variables or click , the **Environment variables for your App** dialog appears. The following tables discusses the three options:

Field	Description
Add	Click Add and the New Environment Variable dialog appears. Enter the variable name and value name here.
Edit	Once an environment variable is added, you can then edit it by clicking Edit.
Remove	Select a variable and click Remove. A confirm dialog appears.

REQUIRED RUNTIMES

When you click **Required Runtimes** or click   , the **Runtimes required by your App** dialog appears. You can pick and choose what runtimes you want to include in your App.

> NOTE: The version attribute must match the installation folder name for the specified version.

The table below lists sample runtimes.

Display Name	Description
.Net .35 runtime plug-in	Plug-in to add a cross OS set of registry keys and files that make up the Microsoft .Net 3.5.
Microsoft J# runtime plug-in	Plug-in to add a cross OS set of registry keys and files that make up the Microsoft J# runtime.
Microsoft Visual C++ 2005 SP1 redistributable runtime plug-in	Plug-in to add a cross OS set of registry keys and files that make up the Microsoft Visual C++ 2005 SP1
Microsoft Visual C++ 2008 SP1 redistributable runtime plug-in	Plug-in to add a cross OS set of registry keys and files that make up the Microsoft Visual 2008 SP1

Once you select one of the default settings, you are prompted to save your changes to the current working file in order to analyze the required runtime changes.

USING THE TEST TASK

Click **Test** to master and test your application locally. This option allows you to master an app file using the given template .tca file. The template .tca file needs to have the files that goes with it. The .tca file has file system information that actually list files for the program it needs to master.

The mastering process verifies the files that are listed in the .tca file data. It then uses these files to complete the mastering of the app file. The template .tca file needs to be a complete and valid .tca file with all the required data (i.e. Publisher, Vendor, etc.) or the mastering fails. The result of this process is the creation of an App OutputName .stp file.

Field	Description
Output Folder	The default output folder is C:\Documents and Set-tings\<user>\Embaracero\Studio\Output.
File Name	The file name that appears is from the information you entered on the Brand panel. It includes the Name and Version of your application. **Note**: Only ASCII characters should be used in File Name field. The mastered application will not launch if non-ASCII characters are used (for example Synérgie).
App	You can select the app file to associated with the .tca file for this application.
Compression	You have three options for determining the file size during mastering: No, Fast, and Best.
Advanced	Allows you select the type of encryption you want to use for the app.
Record Prefetch file information	Selecting this option records the information used to speed up the loading of the application you are mastering. For more information on how to create an app with prefetch, see: Prefetch.
Mastering	After your have set and verified your selections, click **Master**, and the **Mastering App** dialog appears. The mastering process changes a Windows-based application into a streamable format.

App

You can select an app file (.stp) that corresponds to the Studio file you created.

1   Click **Edit** and the **Select a mastered App** dialog appears.
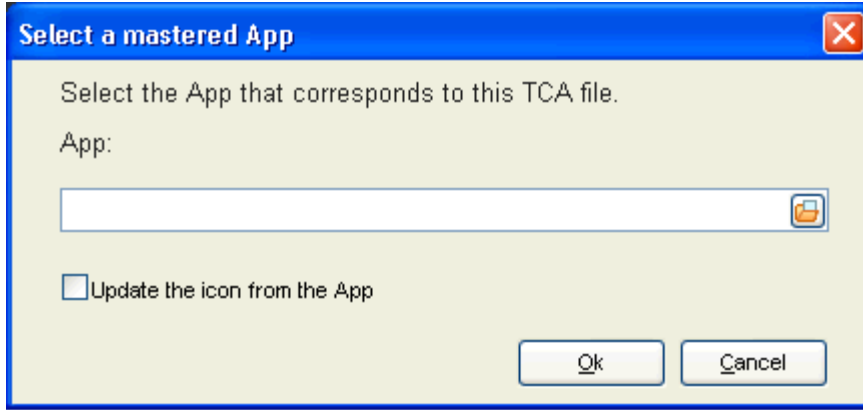
2   Browse to the location of the application's STP File (`c:\Documents and Settings\<user>\Embarcadero\AppWaveStudio\<application .stp>`).

3   You can also choose the option to update the icon from the App.

4   Click **OK** to close the dialog.

STP File

This file is the outer layer that encapsulates the other application components. These files are stored in `c:\Documents and Settings\<user>\Embarcadero\Product Repository\<application\<version number>\<product name and version>\`. The other application components include:

| File Extension	Description
aib	This file contains the metadata that describes the content (or "blueprint" of the STC file.
stc	The code and other configuration information that comprises the streaming content of an application.
tok	Also known as a virtualization token, token files are comprised of security data.
ico	icon file.
.htm/.txt	EULA file.

Compression

You have three options for determining the file size during mastering:

- No: This is useful for testing, but should not be used for streaming.

- Fast: Compresses and decompresses quickly. Use this setting to reduce mastering times. Can be used to stream on faster networks.

- Best: Decompresses faster, but it takes longer to compress. This method is recommended because it results in a 20% smaller App. This results in a greater transfer rate when streaming.

Advanced

The Advanced Button allows you select the type of encryption you want to use for the App.
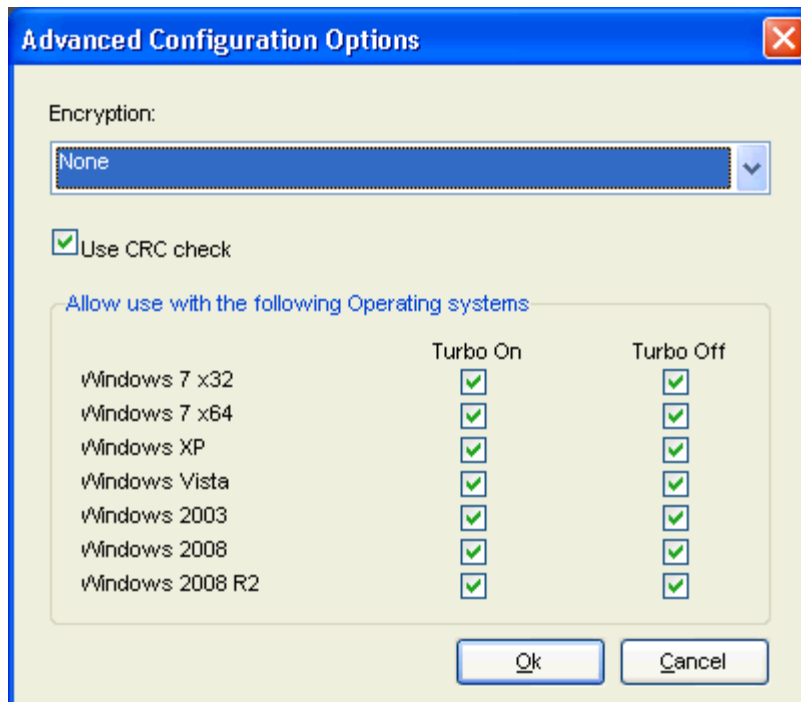
- This encryption provides a high degree of security, ensuring that application data remains safe and private when streaming over the Internet. You have three encryption options:

  - None: No encryption is needed.

  - Bitshift: An encryption method that shifts the digits to the left or the right.

  - AES_CFB: Cipher Feedback mode which makes a block cipher into a self-synchronizing stream cipher.

- Selecting the **Use CRC check** option specifies that the app integrity is verified based on the md5 file, in order to detect possible transfer data error.

- Under the **Allow Use within the Following Operating Systems** option, when the Turbo On option is selected, it indicates that the app is allowed to run on the specified OS and in Turbo mode. When deselected, the check box prevents the app from launching on the specified OS and in Turbo mode.
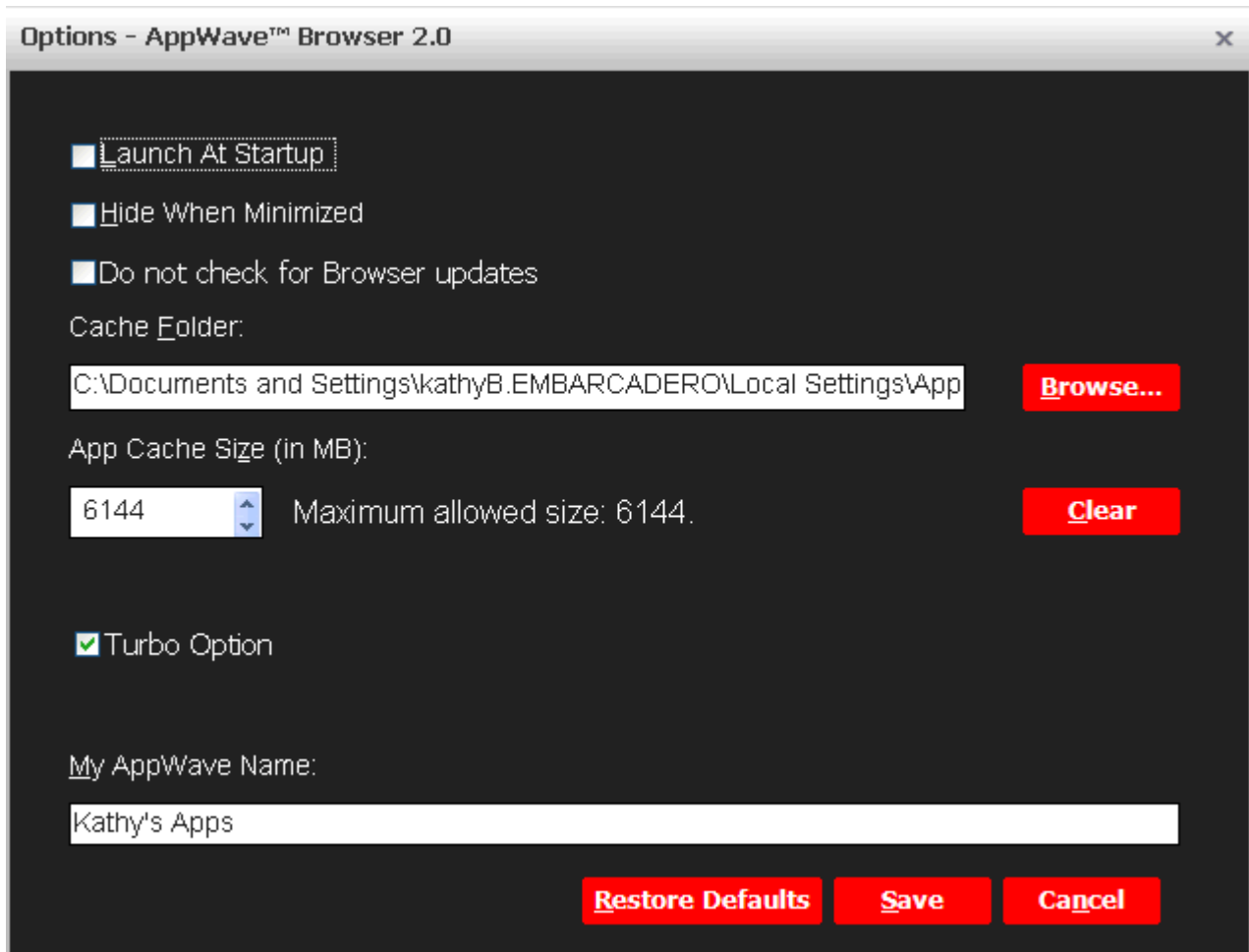
Prefetch

Prefetch is the process of streaming application blocks in anticipation that those blocks are needed by an application.

> NOTE: Only some key blocks of application data from AppWave are necessary for the application to launch and perform basic functions.

To create an app with prefetch:

1 Before you start mastering an app with prefetch, it is recommend that you save the .tca file under a different name, for example xxx_base .tca, to ensure that you still have the original base project.

   • Mastering with prefetch requires that you are using the Turbo Option, which means you'll need to login on a VM as an Administrator.

To see if the Turbo Option is selected, click **Tools and Options** (   ) **> Options**. The Turbo Option is selected by default.

2 Setup and record the application as you normally would. For more information on the steps required, see [Create AppWave Supported or Custom App from Install.](#)

3   On the **Test** page, make sure that **Record Prefetch file information** is selected. If this option is disabled, you are not running in Turbo mode.
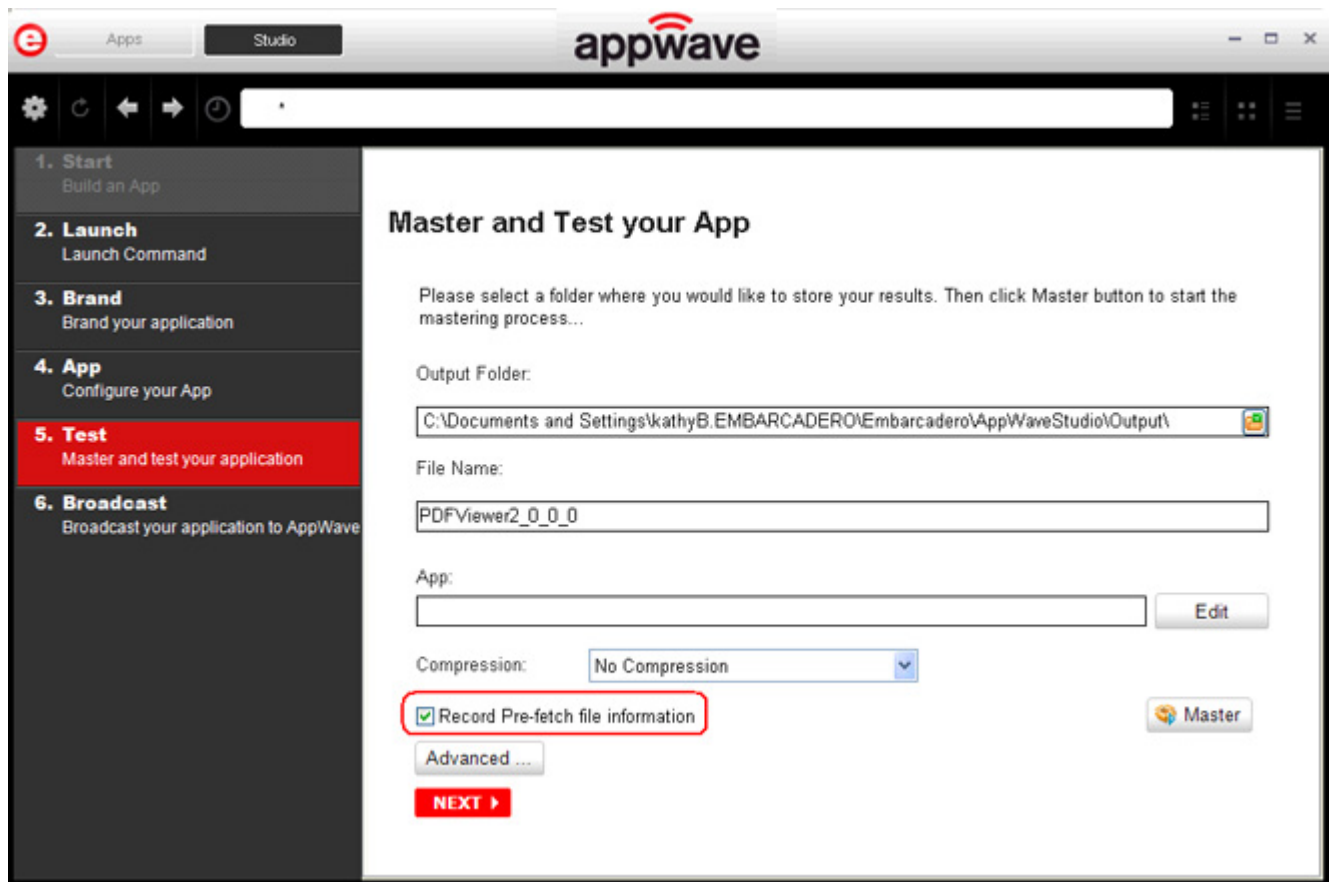
4   On the Test page, click **Master** (Master). You are prompted to save your file.

5   Once the mastering is complete, click **Launch** instead of clicking **Finish**.
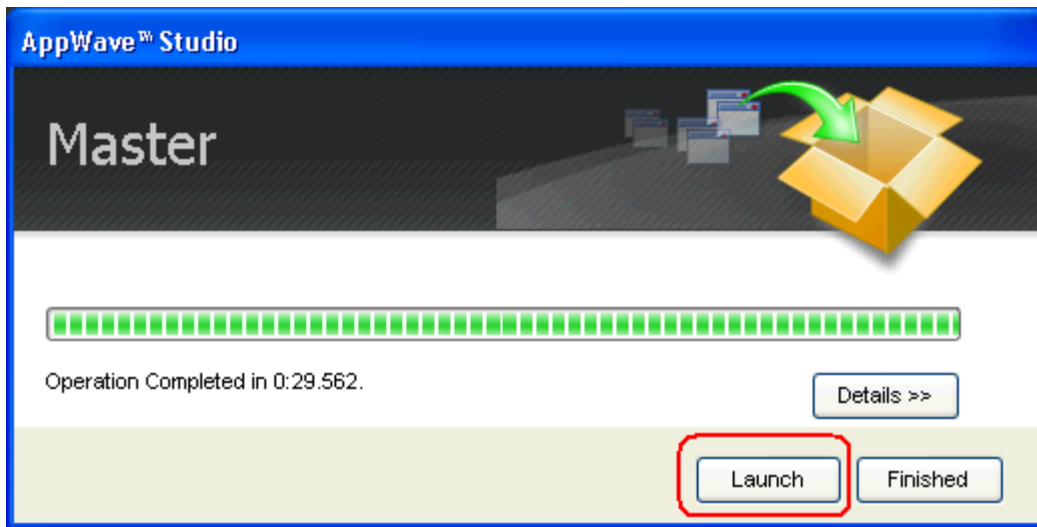
6   When the application starts, the prefetch recording also starts.

NOTE: The prefetch recording happens in the background and is not visible to the user.

7   Close the application once it has launched. Closing the application stops the prefetch recording.

8   Once you have closed the application and the prefetch recording, the next step is to remaster the application:

• On the **Test** page make sure that the **Record Prefetch file information** option is no longer selected.

NOTE: If you choose to leave it selected, after mastering with prefetch click **Finish**, instead of **Launch**.

9   Open the **Broadcast** page and make sure the option to **Always remaster projects before broadcasting to the AppWave** is selected.

10   Click **Broadcast** and save the changes to your file. The **Broadcast** dialog appears where you can track the remastering and broadcast of your app with the prefetch recorded information included.

11   Your app is now mastered with the prefetch information included.

USING THE BROADCAST TASK

The last step is to broadcast your application to the AppWave.

Field	Description
AppWave URL	The default URL appears in this box. You can edit this URL.
Always remaster projects before broadcasting to the AppWave	This is selected by default. This is to guarantee that the app has the latest updates.
Broadcast	Click **Broadcast** and a dialog appears telling you need to save your file before continuing. Click **OK** and the Broadcasting dialog appears. Save the file in *C:\Documents and Settings\<username>\Embarcadero\AppWaveStudio\Output directory*.

MASTERING

This section is an overview of all the steps involved in mastering an application to create an app executable. Successful mastering depends on a thorough understanding of the components of each element. app executables provide one-step functionality for each product. This simplifies broadcasting and enables side-by-side versioning of products.

You can master an app file using the given template Studio file. This file needs to also have the files associated with it.

For more information, see:

- Understanding the Studio File

- Before you Master an Application

- Obtaining Mastering Support

- What is Required for App mastering

- Using the Command Line

UNDERSTANDING THE STUDIO FILE

The app file is mastered using the Studio or .tca file. A .tca file is a set of information that defines the resources, launch points, and registry settings required to run a product. It also contains information about the product such as the version, name, and editions. The .tca file needs to have the files that goes with it and has a file system information that actually list files for the program it needs to master. The files have to be physically present at the directory information that is saved in the .tca data.

The mastering process verifies the files that are listed in the .tca file data. It then uses these files to complete the mastering of the app file. The template .tca file needs to be a complete and valid .tca file with all the required data (i.e. Publisher, Vendor, etc.) or the mastering fails. The result of this process is the creation of an App OutputName .stp file. All the listed arguments are required. The OutputLocation directory needs to have a drive specified, and if the directory does not exist then the program tries to create one. The OutputName argument is used as a name for the .stp file, so there should not be any file named OutputName .stp in the OutputLocation directory.

BEFORE YOU MASTER AN APPLICATION

- Understand the application

- Know what registry keys are used by the application

- Know what folders are used by the application

- Know what root folders or registry keys were created

- Know where the master files (the App) and the Studio Files (files used during application creation) are located

- Understand how the application is licensed

- Understand the prerequisites

- Understand the scripting needs

OBTAINING MASTERING SUPPORT

When you want to master an application, AppWave Studio will:

1 Retrieve the AppWave file (includes brand, launch, required plugins, EULA, and Icons) from the AppWave. This information is used to automatically populate the product information;

2 Ensure that the latest version of all required plugins are available;

3 Download a plugin if it is missing;

4 Compare the existing plugin with the version on the mastering computer to ensure the latest version is used.

WHAT IS REQUIRED FOR APP MASTERING

Creating an app involves:

- Cataloging the product in the Studio, including branding and all the data that makes up the studio metadata file.

- Having the EULA and Icons for the application.

- Have the plugins that modify captured studio metadata file (.tca) so that the product is portable and runs on other platforms.

- Prerequisites, those things that are required by the product so the master runs on systems that don't have the prerequisites.

For more information, see:

- [Application Types](#)

- [File Formats](#)

Application Types

AppWave works with many types of applications. Two of the most typical are:

- Open source products or companies that allow distribution. These mastered applications can be redistributed as apps.

- Products that can't be redistributed. You then rely on instant mastering. Once the product is mastered and placed in your AppWave you do not have to master it again.

File Formats

Two file formats are used to add Applications and app mastering support for the AppWave and AppWave Browser.

- .PADE is a plugin used to help users master an application by automatically manipulating the file and registry entries in the virtual application

- .AW is an AppWave file that adds Instant mastering support to a server and browser

  The AppWave file (.AW), created by the Plugin Manager, is what is used to add Instant mastering support to a server. The utility also allows the user to choose which plugins are required for Instant mastering of the target application. The AW file is saved with the same name as the APP.

  NOTE: The Product Manager is located in the directory as the AppWaveCommand Line and the AppWaveBrowser executable.

USING THE COMMAND LINE

In order to use this option the user must open in CMD C:\Documents and Settings\Administrator\Local Settings\Application Data\Embarcadero\ToolBox\TCAcomandline.exe.

The available options are:

Option	Description
AppWavec -B	Broadcast file(s) to an AppWave
AppWavec -X	Creates .app from .tca and .stp

Option	Description
AppWavec -PPL	Creates .pade from a plugin
AppWavec -A	Show advanced option

1   Broadcast file(s) to an AppWave: The -B or -Broadcast option broadcasts to an AppWave one of four different ways: 1) an .App file; 2) a Studio/STP file combination; 3) a PADE (masteered plugin) file; or 4) an AW file. For the second item, the Studio file has to be the file used to generate the STP file.

   • Usage 1: Broadcast a project (Studio and STP files are required) AppWavec (-B|-Broadcst) (-T|-StudioFile) <myProduct .tca> -STP <myProduct .stp> -URL <AppWave> -U [<domain>]<user> [-PWD <password>] [-EDN] [-Debug]*

   • Usage 2: Broadcast an App (.app file is required)AppWavec (-B|-Broadcast) -APP <myProduct .app> -URL <AppWave> -U [<domain>]<user> [-PWD <password>] [-EDN] [-Debug]

   • Usage 3: Broadcast PADE plugins (.pade files)AppWavec (-B|-Broadcast) -PADE <myPlugin .pade> -URL <AppWave> -U [<domain>]<user> [-PWD <password>] [-EDN][-Debug]

   • Usage 4: Broadcast AppWave templates (*.aw files) AppWavec (-B|-Broadcast) -AW <myAppWave .aw> -URL <AppWave> -U [<domain>]<user> [-PWD <password>] [-EDN] [-Debug]

   • Argument descriptions:

Argument	Description
-B or -Broadcast	Selects the broadcast option.
-T or -StudioFile	The full path and file name of myProduct's .TCA file.
-STP	The full path and file name of myProduct's .STP file (created during mastering).
-APP	full path and file name of the mastered myProduct's .APP file.
-PADE	Full path and file name of the mastered plugin's .PADE file.
-AW	Full path and file name of the AppWave file to be broadcast.
-URL	The AppWave URL.
-U or -UserName	The UserName for the AppWave.
-PWD	The password for the user (optional).
-EDN	Login with public\Embarcadero Developers Network account.

Argument	Description
-Debug	To output debug messages.
myProduct .tca	The full path and filename of your product's Studio file.
myProduct .stp	Full path to the STP file created when mastering myProduct.
myProduct .app	Full path to the APP file created when broadcasting myProduct.
myPlugin .pade	Full path to the PADE (mastered plugin) file.
AppWave	The AppWave URL.
user	The username or domain\username for the AppWave login.
password	The password for the AppWave.

2   Export: This option exports an App file (creates an APP file from a Studio/STP file combination. The Studio file has to be the file used to generate the STP file.

   • Usage: Create *.app file from TCA and STP:AppWavec (-X|-Export) -T <myProduct .tca> -STP <myProduct .stp> -O <outputLocation> -N <outputName> [-Debug]

   • Argument descriptions:

Argument	Description
-X or -Export	Selects the export option.
-T or -StudioFile	The full path and file name of the template Studio file.
-ST{	The full path and file name of the mastered App.
-O or -Out	The file output directory.
-N or -Name	The output file name.
-Debug	To output debug messages.
myProduct .tca	The full path and filename of the template Studio file.
myProduct .stp	The full path and filename of the app file to be broadcast.
outputLocation	The full path of the directory to save to.

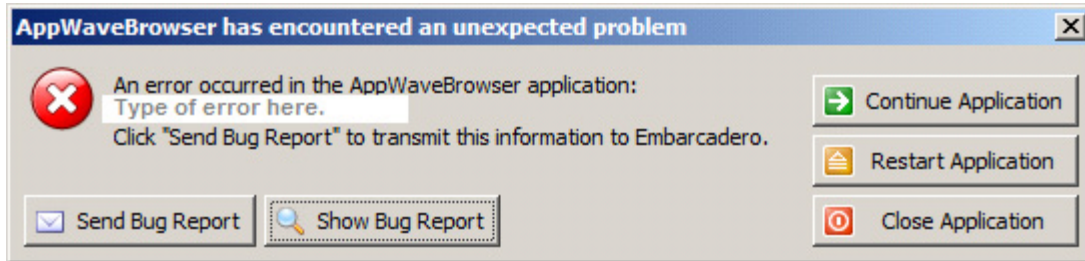| Argument	Description
outputName	The name of the app file to create.

3   Creates pade from a plugin: The -PPL option packages the files of a PADE plugin into a PADE file. It requires the path to the folder containing the plugin files. Two files must be included: the version .dat and the main plugin file that has the same name as the plugin. Optionally additional files may be included. Only the files listed in version .dat will be included in the PADE file.

- Usage: Generate *.pade filesAppWavec -PPL (-PL|-Plugin) <fullPathToPluginFolder> [(-O|-Out) <outputLocation>] [-Debug]

- Argument descriptions:

Argument	Description
-PPL	Selects the package plugin option
-PL or -Plugin	The full path to the folder containing the plugin files.
-O or -Out	The full path of the output location directory.

4   Show advanced option: The -PPL option packages the files of a PADE plugin into a PADE file. It requires the path to the folder containing the plugin files. Two files must be included: the version .dat and the main plugin file that has the same name as the plugin. Optionally additional files may be included. Only the files listed in version .dat will be included in the PADE file.

- Usage:AppWavec -PPL (-PL|-Plugin) <fullPathToPluginFolder> [(-O|-Out) <outputLocation>] [-Debug]

- Argument descriptions:

Argument	Description
-PPL	Selects the master plugin option.
-PL or -Plugin	The full path to the folder containing the plugin files.
-O or -Out	The full path of the output location directory.

AppWave Studio Error Dialog

The AppWave Browser encounters a problem the error dialog appears.

This dialog has the following actions that can be taken.

Element	Description
Continue Application	This option is available if an error did not close the application.
Restart Application	An error closed the AppWave Browser. Click Restart. If you were creating an App, that process was terminated.
Close Application	Closes the AppWave Browser application. This action does give you a warning dialog.
Send Bug Report	Opens a Send Assist dialog where you can enter your name, email, and details about the error.
Show Bug Report	Opens the AppWaveBrowser .exe dialog which contains the Bug Reports.

Bug Reports

When you click Show Bug Report in the Error dialog the AppWaveBrowser .exe dialog appears. There are nine different tabs showing information on the bug types.

- General
- Call Stacks
- Modules
- Processes
- Hardware
- CPU Regs
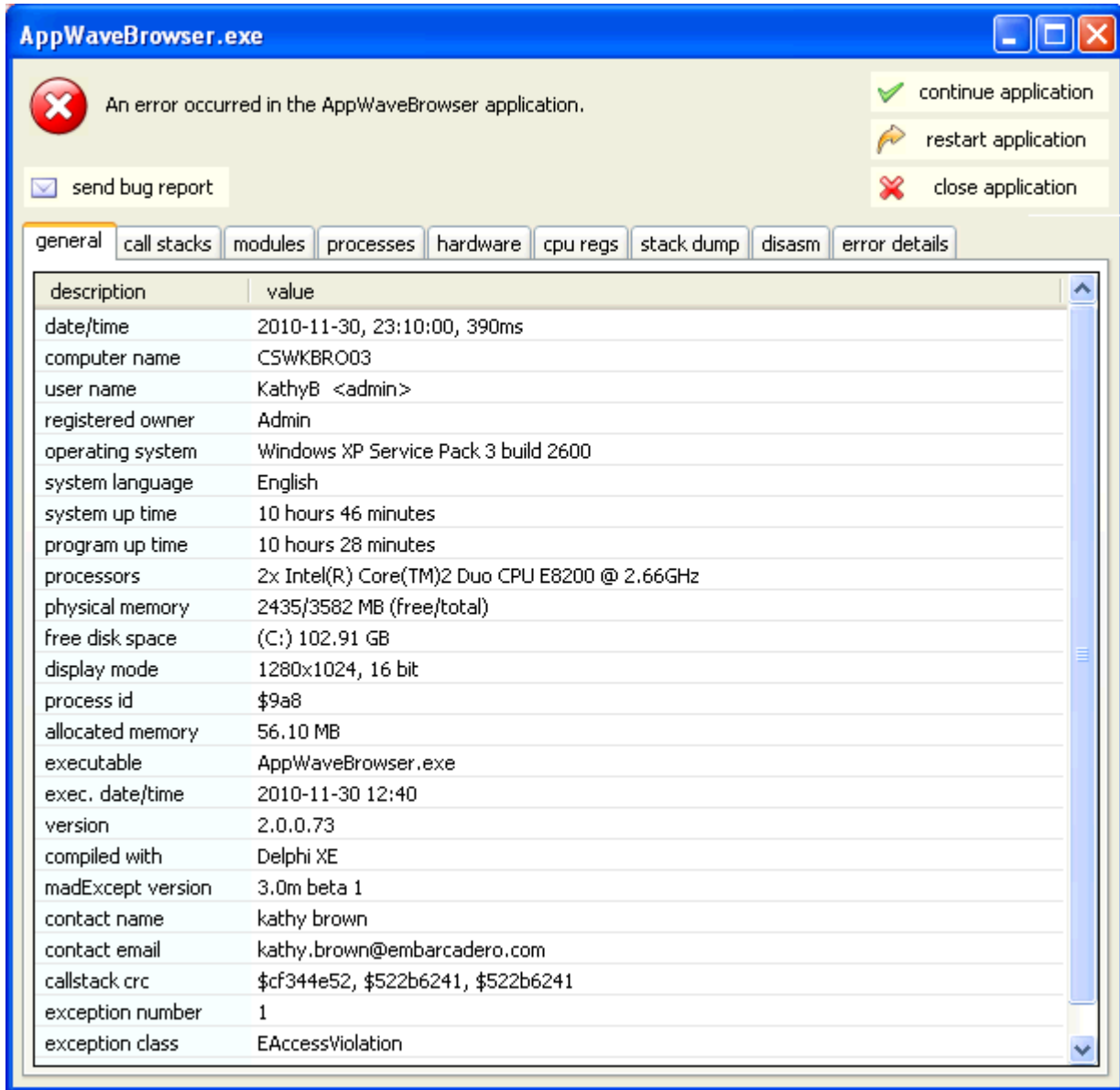
- [Stack Dump](#)
- [Disasm](#)
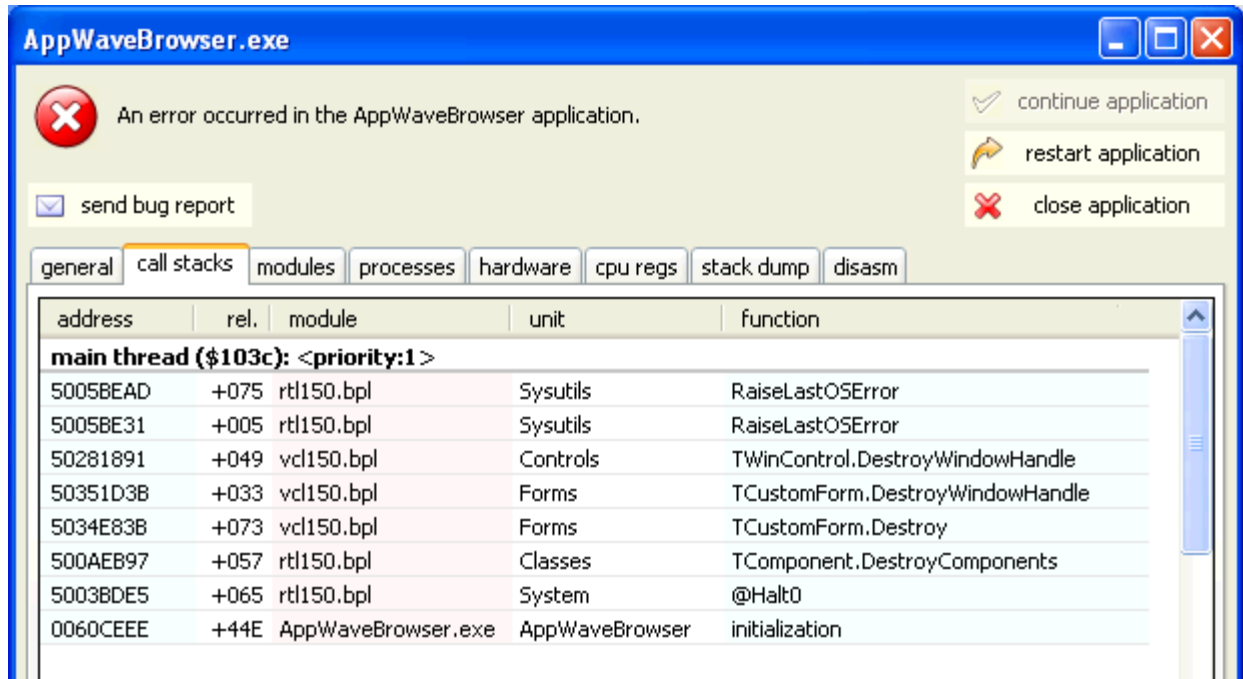
GENERAL
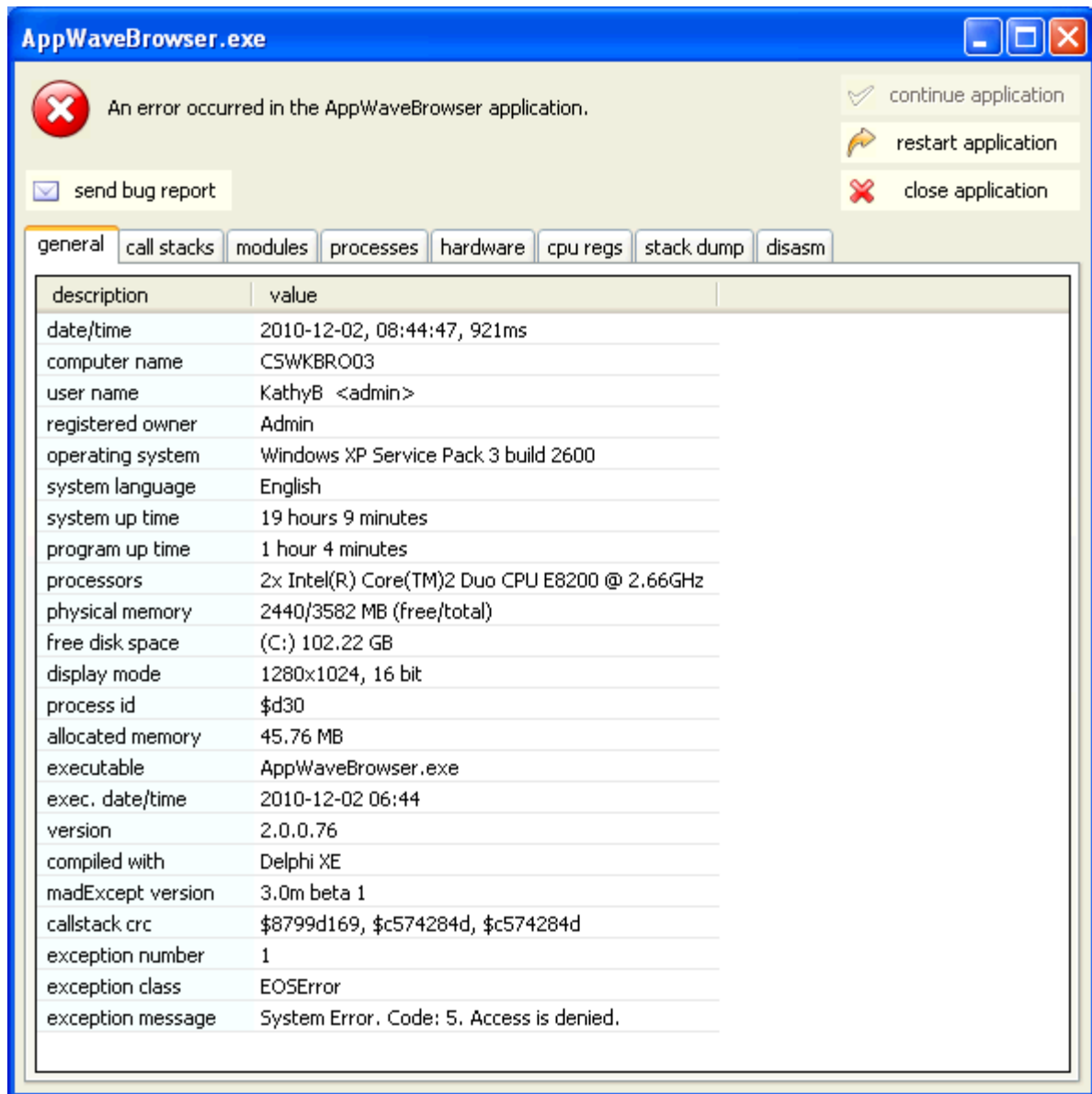
CALL STACKS
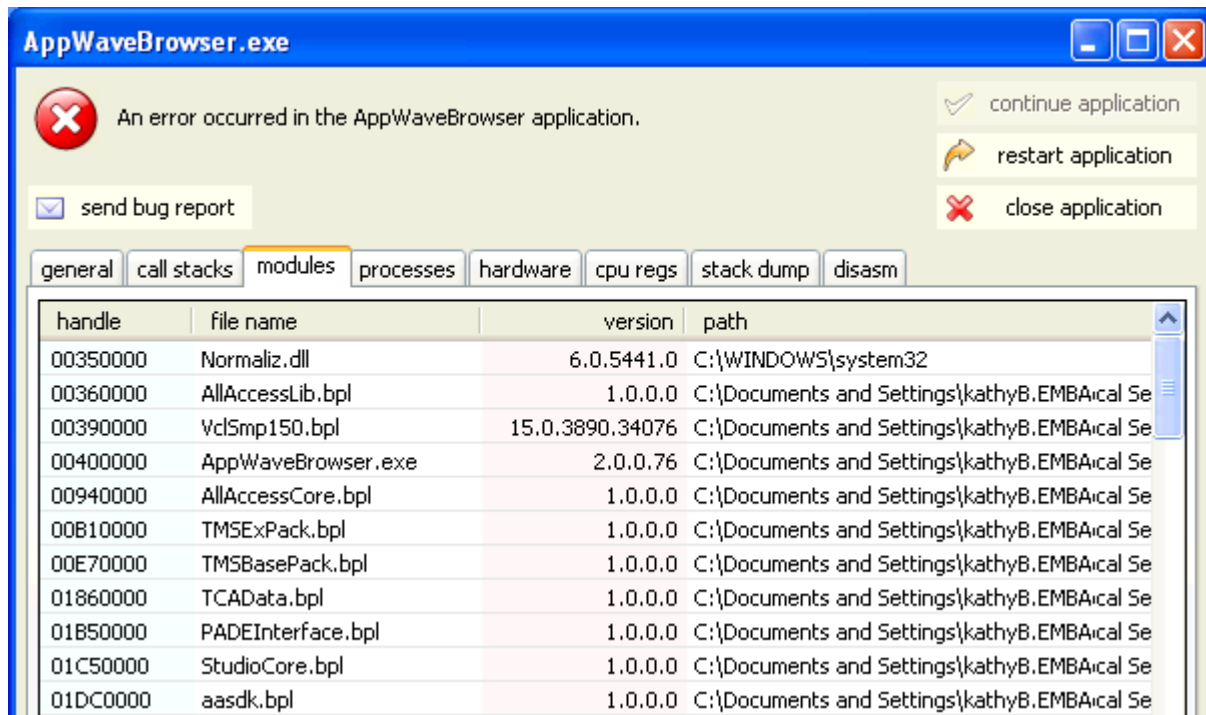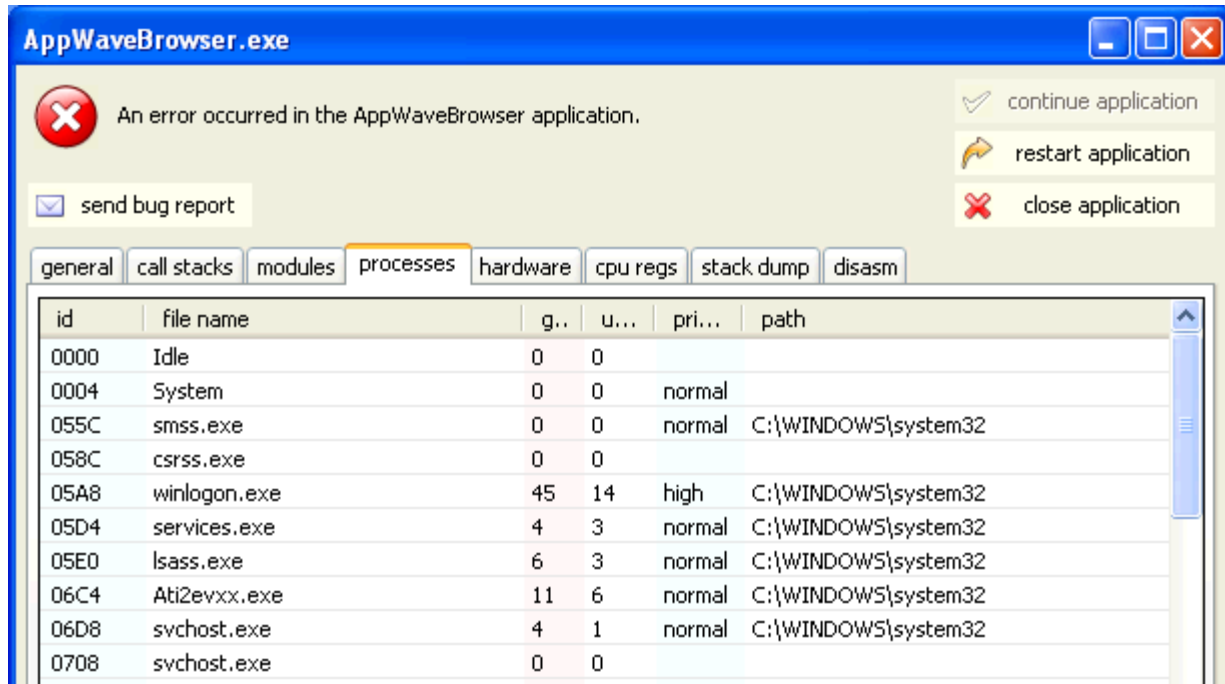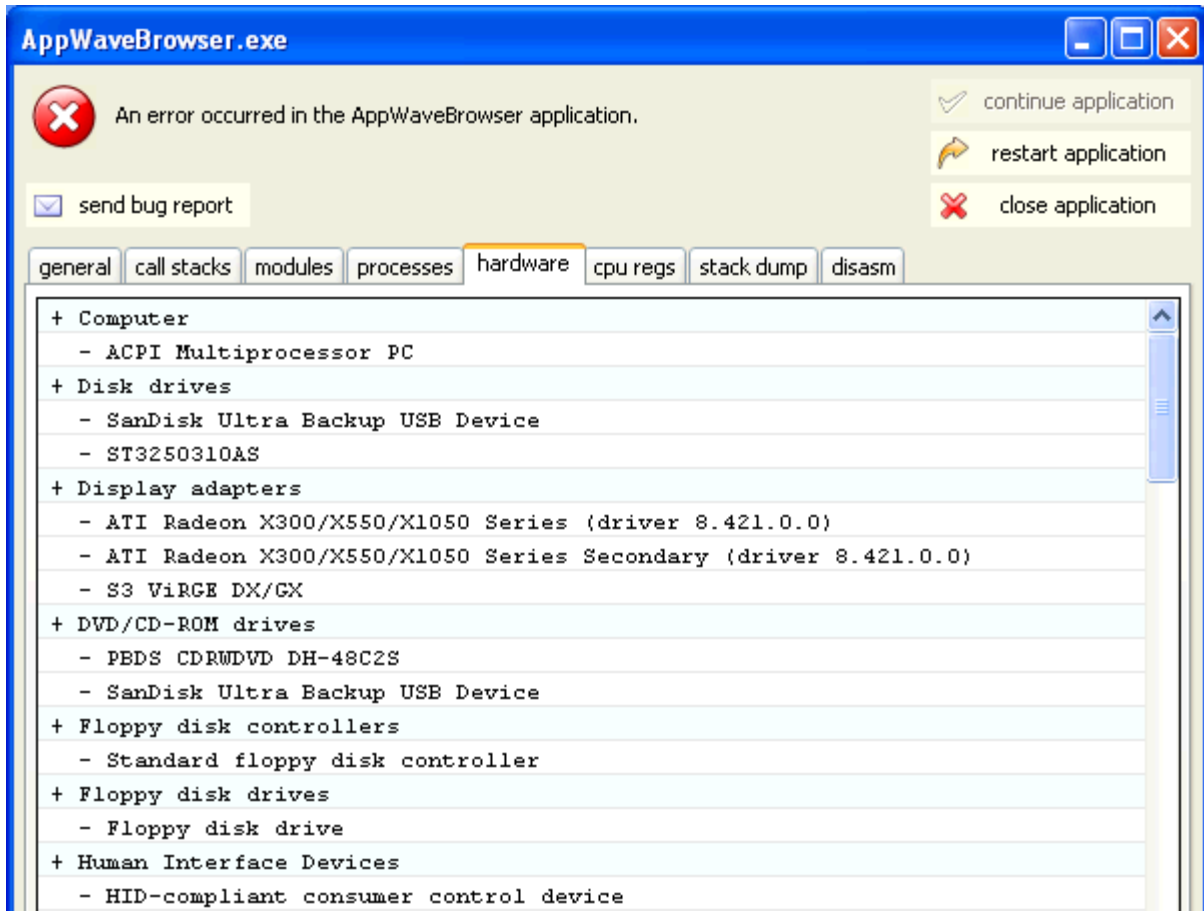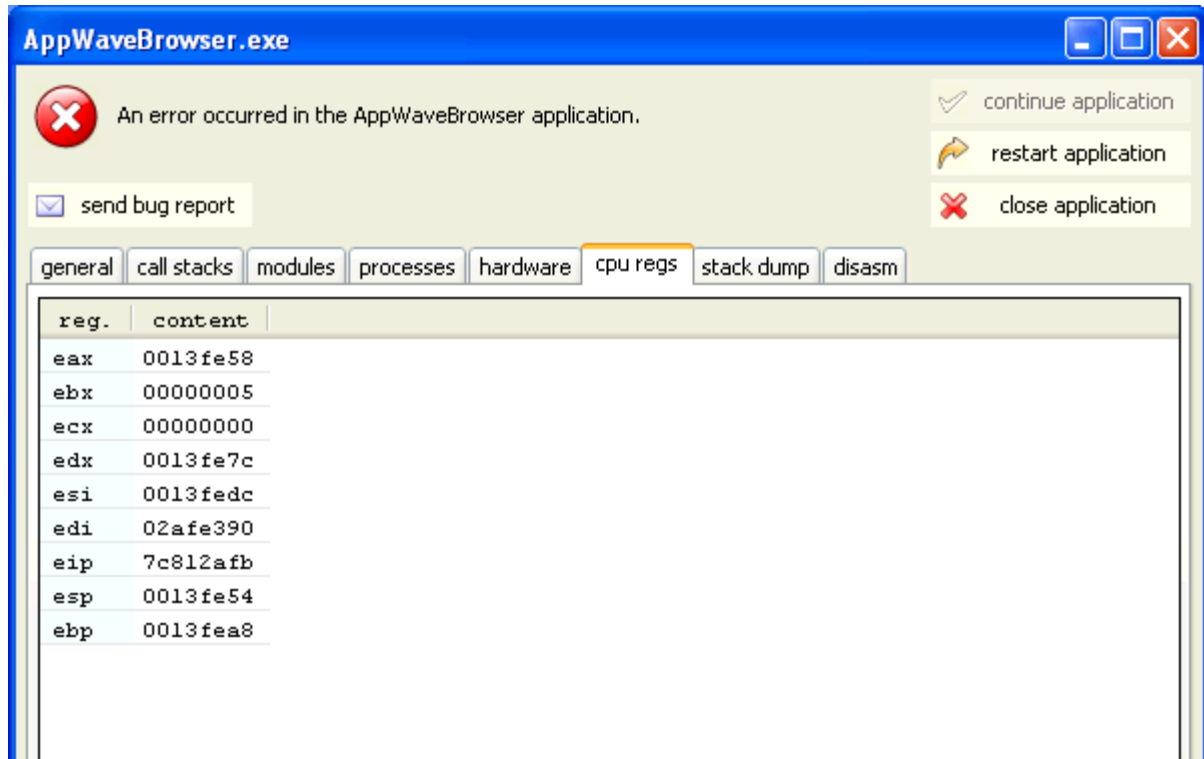

MODULES

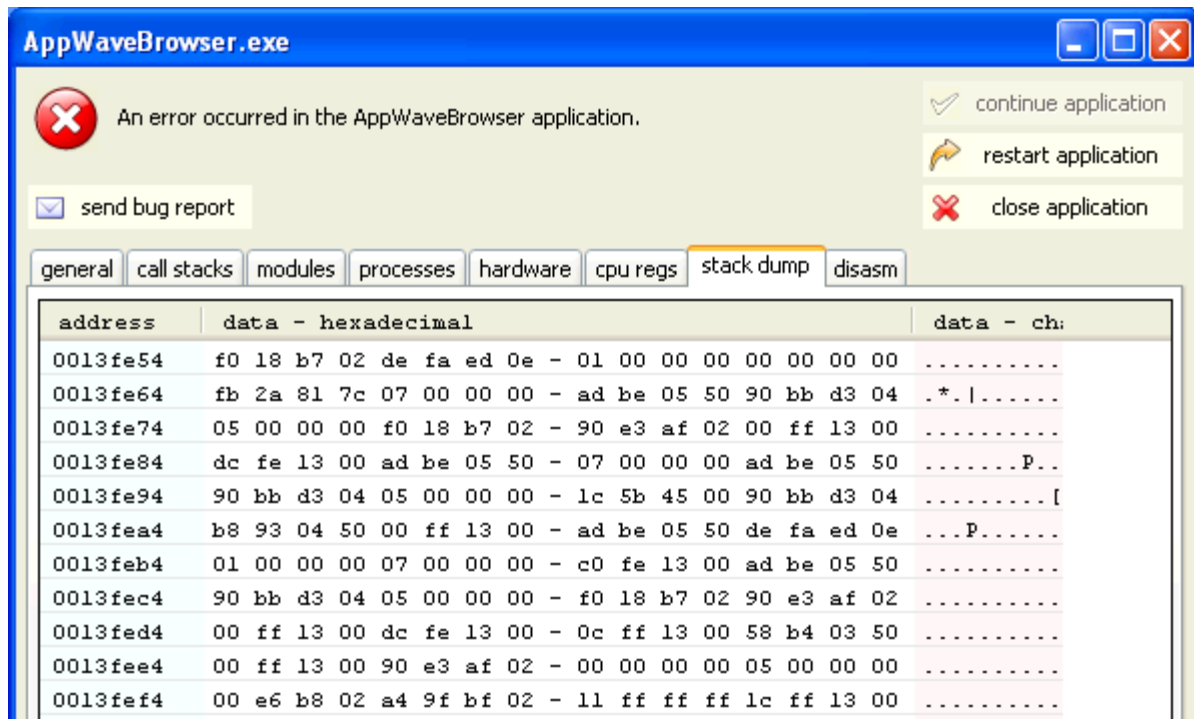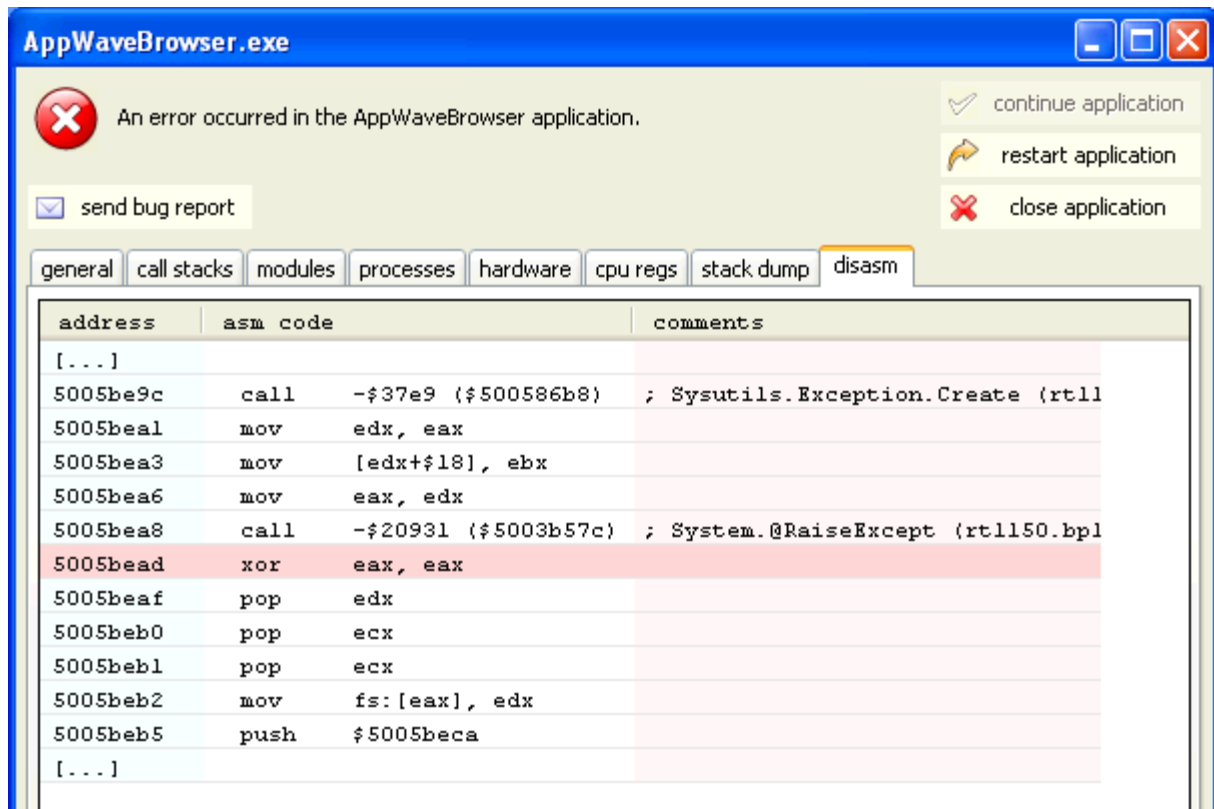## PROCESSES

HARDWARE

CPU Regs

STACK DUMP

DISASM

This bug report displays malicious behavior or vulnerable code in binary files.

AppWave Studio Appendix

Install and Uninstall Folder "CleanUp"

When you uninstall and reinstall AppWave Studio, at times it is recommended o fully remove all the files and directories installed by AppWave. This ensures a clean starting place for your next install.

Following files should be removed for the AppWave Studio uninstall:

Windows XP:

- C:\Documents and Settings\<user>\Local Settings\Application Data\Embarcadero\AppWaveBrowser\plugins

- C:\Documents and Settings\<user>\ AppWaveStudio\Output\ - the default Output folder path *can be changed by the user*

Windows 7:

- c:\Users\<user>\AppData\Local\Embarcadero\AppWaveBrowser\plugins\

- c:\Users\<user>\Embarcadero\AppWaveStudio\Output\ - this directory *can be changed by the user*

AppWave:

- Install directory -> c:\Program Files (x86)\Embarcadero\AppWave

- Application repository, only if you want to delete existing binaries -> c:\ProgramData\Embarcadero AppWave\Products

The following files should be removed for the AppWave Browser uninstall:

On Windows XP:

- Install directory - "C:\Documents and Settings\<username>\Local Settings\Application Data\Embarcadero\AppWaveBrowser"

- Repository - "C:\Documents and Settings\<username>\Local Settings\Application Data\Embarcadero\Product Repository"

- Streaming Core additional locations:
    - "C:\Documents and Settings\<username>\Application Data\Embarcadero\Product Repository"
    - "C:\Documents and Settings\All Users\Application Data\Embarcadero\StreamingCore"

On Windows 7:

- Install directory - "C:\Users\<username>\AppData\Local\Embarcadero\AppWaveBrowser"

- Repository - "C:\Users\<username>\AppData\Local\Embarcadero\Product Repository"

- Streaming Core additional locations:
    - "C:\Users\<username>\AppData\Roaming\Embarcadero\Product Repository"
    - "C:\Users\All Users\Embarcadero\Product Repository"
    - "C:\Users\All Users\Embarcadero\StreamingCore"

FILES

The files listed below are the mandatory files to be delivered in the app process.

App File

.app is the filename extension for AppWave-Enabled application files. It is an application that was mastered with AppWave and must adhere all of the App Mastering Acceptance Criteria.

App means that applications can be streamed from the AppWave and then run locally on the desktop without installation. This options also allows applications to be run on locked-down desktops. There are no conflicts with other applications, no writing to the system areas of their registry or modifying shared dlls. Application data such as Configuration and license information is stored locally on the hard disk, and is separate from the executable itself.

.aw File

The AppWave Template file offers a model for mastering an application, and contains branding information and information on which filter to run on the installer for the application. It must respect all of the "Acceptance criteria".

The AppWave Template file (.AW) is created by the PADE Product Manager and it used to add Instant AppWave Mastering support to a server.

MD5 File

Message-Digest algorithm 5 (MD5) is used to check the integrity of files. The .aw and the .app files must come together with the associated md5 message digest file. Each time one of the two files needs to be changed, the .md5 file must be changed too.

.msi installer File

MSI files are database files, used by Windows Installer. They contain information about an application which is divided into features and components, and every component may hold files, registry data, shortcuts etc. The MSI file also contains the UI that is to be used for installing, and various other data such as any prerequisites to look for, custom actions to execute, the order of the installation procedure, whether to support Administrative installations, etc. MSI files are the current recommended way of doing installations on Windows.

The .msi files provided must have the ability to run in Unattended mode.

.stp File

STP File is created after the author masters the application and can be found in the output folder. This file is the outer layer that encapsulates the other application components. It is encrypted and stores the files, folders, and registry settings needed to reproduce the application being mastered on the end users machine.

The .stp file includes:

File Type	Description
Aib (application installation blueprint)	This file contains all the files' metadata (filename, file path, file size, etc) and registry information (registry keys and values) for the mastered application.
Stc (streaming content file)	The code and other configuration information that comprises most of the application.
Tok (token)	Token Files are comprised of security data and general information about the mastered application
Ico	icon file
htm	EULA file

None of the files contained in the .stp file need to be altered or deleted.

Studio File

The Studio or .tca file is the project file in the AppWave Browser.The .tca contains the registry and folder and files structure captured during a software installation along with resources, launch points, and registry settings required to run a product.It also contains information about the product such as the version, name, and editions. Every time a user needs to master an App, a .tca file is created.

LOG FILES

Log files are the text files which contains all the log entries saved by AppWave and AppWave Browser.

AppWave Browser logs:

- StudioLog.log

- StreamingCore.log

- log4delphi.log

AppWave logs:

- aaserver.log

- error.log

- info.log

Log path:

- C:\Documents and Settings\<user>\Local Settings\Application Data\Embarcadero

- C:\Documents and Settings\<user>\Local Settings\Application Data\Embarcadero\Product Repository\StreamingCore

- C:\Documents and Settings\<user>\Local Settings\Application Data\Embarcadero\AppWaveBrowser

- C:\Program Files (x86)\Embarcadero\AppWave\2.0\logs

Listed below are logs that apply to AppWave Launcher (AWAPP) as well as their file location.

1 C:\Documents and Settings\ Administrator \Local Settings\Application Data\Embarcadero\AppWaveBrowser\StudioLog.log

2 C:\Documents and Settings\ Administrator \Local Settings\Application Data\Embarcadero\StreamingCore\Log\StreamingCore.log

3 C:\Documents and Settings\Administrator\Local Settings\Application Data\Embarcadero\Product Repository\StreamingCore\Profiles\Administrator\Sessions\0\Applications{855675B0-84F8-4A94-AE29-D7B2CB849AB7}\FSD\Private\C\AppWaveHost20\logs\aaserver.log

4 C:\Documents and Settings\Administrator\Local Settings\Application Data\Embarcadero\Product Repository\StreamingCore\Profiles\Administrator\Sessions\0\Applications{855675B0-84F8-4A94-AE29-D7B2CB849AB7}\FSD\Private\C\AppWaveHost20\elc\logs\error.log

5 C:\Documents and Settings\Administrator\Local Settings\Application Data\Embarcadero\Product Repository\StreamingCore\Profiles\Administrator\Sessions\0\Applications{855675B0-84F8-4A94-AE29-D7B2CB849AB7}\FSD\Private\C\AppWaveHost20\elc\logs\error.log"

6   "C:\Documents and Settings\Administrator\Local Settings\Application Data\Embarcadero\Product Repository\StreamingCore\Profiles\Administrator\Sessions\0\Applications{855675B0-84F8-4A94-AE29-D7B2CB849AB7}\FSD\Private\C\AppWaveHost20\elc\logs\info.log

7   C:\Documents and Settings\All Users\Application Data\Embarcadero AppWave\logs\ion.log

APP MASTERING ACCEPTANCE CRITERIA

Required Mastering Criteria

Mandatory Fields

Conditionally Required Fields

Recommended Folders

Required Mastering Criteria

1   No Beta Versions of applications are accepted.

2   All APPs must be mastered using the [Prefetch](#) option.

3   The vendor must provide the .msi installer of the application being mastered; If the application has an installer with any other file extension, then the vendor must convert that installer into an unattended .msi installer and provide that one. The vendor has the obligation to check that the converted installer works properly on Windows XP, Windows 7 32-bit and Windows 7 -64 bit Operating Systems.

4   The app-mastering process must always be done with the original installer of the application and not with the converted one.

Mandatory Fields

1   Product Name along with the trademark sign

2   Product Display Name along with the trademark sign

3   Product Version

  • If no Product Version is specified in the installer then it will be filled in with the default value "1.0".

  • Product Version must not contain build number. Format accepted is a.b.c.d (c and d values are optional). For example valid versions would be: 1.0, 1.0.1, 1.0.1.1. Invalid Versions would be: 1.0.1234, 1.0.1.12345 (should be modified to be 1.0 and 1.0.1).

4   Publisher

5   Product Website

6   EULA file(s) (in .htm format with Arial font and size 10); If the application requires prerequisites (like dotNet Framework or Adobe AIR) then the correspondent EULA file should be included as well.

7   A screen shot of the main window of the application, which must be in .png format. The screen shots will be scaled to 228 x 174 pixels and must not be distorted. The images used must not be under copyright.

8   Product description (cannot be copyrighted material).

9   Long description (cannot be copyrighted material).

10   Product Icon (must be in .png file format).

11   Product release date (if not specified it will be the current date).

12   License information.

Conditionally Required Fields

These are fields that must be completed if the information is available.

1   System Requirements

2   Support Contacts

3   Edition

Recommended Folders

When mastering an application, it is best practise to install in a folder not is not on a network share.

TEMPLATE ACCEPTANCE CRITERIA

1   The app executable used to build the template must pass all the App Mastering Acceptance Criteria listed in the App Mastering Acceptance Criteria.

2   All .apps produced with the template delivered must launch on:

   • Windows 7 32-bit

   • Windows XP 32-bit

   • Windows 7 64-bit,

   NOTE: No matter on which OS the .app was created, the template must work in the same manner on Windows 7 and Windows XP.

3   Naming standard:

- The .aw file name must be set to productname.aw.

- The File Name must be lowercase characters only.

- Only ASCII characters should be used in File Name field. The mastered application does not launch if non-ASCII characters are used (for example Synérgie).

4   All the necessary plugins must be included in template file.

>