

---

# Amazon WorkSpaces Application Manager **Administrator's Guide**





## **Amazon WorkSpaces Application Manager: Administrator's Guide**

Copyright © 2016 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

## Table of Contents

What is Amazon WAM .....	1
Available Regions .....	1
Related Services .....	1
Setting Up .....	2
AWS Account .....	2
Amazon WorkSpaces .....	2
Controlling Access .....	3
Create the Application Packaging Role .....	3
Create the AWS Marketplace Access Role .....	3
(Optional) Grant an IAM User Access to Amazon WAM .....	4
Getting Started .....	5
Sign Up for Amazon WAM .....	5
Build Your Application Catalog .....	6
Add an AWS Marketplace Application .....	6
Add Your Application .....	7
Assign Applications to Your Users .....	8
Amazon WorkSpaces Application Manager Client Application .....	8
Managing Applications .....	9
Building Your Application Catalog .....	9
Subscribing to an Application .....	9
Uploading Your Own Applications .....	10
Removing an Application .....	11
Assigning an Application .....	13
Updating an Application .....	14
Versions .....	15
Constraints .....	16
Tags .....	16
Users and Groups .....	16
Support Details .....	16
Monitoring Application Usage .....	17
Applications .....	17
Users .....	17
Removing All Application Assignments .....	17
Troubleshooting .....	18
I assigned an application to a user, but I don't see the user in the usage report .....	18
I don't know how to manage User Access Control on Windows .....	18
I am performing HTTPS Proxy Inspection on my WorkSpaces .....	19
Packaging and Validating .....	20
Packaging an Application .....	20
Best Practices .....	21
Launching the Packaging Instance .....	21
Creating a New Package .....	21
Creating a New Application Version .....	27
Understanding Color Coding .....	27
Packaging Specific Applications .....	30
Using Licenses .....	34
Validating a Package .....	35
Launching the Validation Instance .....	36
Validating the Application Package .....	36
Updating a Package .....	37
Working with Files and Folders .....	37
Working With the Registry .....	41
Working with Environment Variables .....	43
Working with Fonts .....	44
Working with Startup Items .....	46

Working with Services and Drivers .....	48
Adding Required Applications or Scripts .....	50
Sandboxing Your Package .....	53
Setting File Security .....	55
Finding a Package .....	56
Sharing a Package .....	56
Deleting a Package .....	57
Troubleshooting .....	57
Finding the Log Files .....	58
Application Packaging Issues .....	58
Device Driver and Service Issues .....	61
Missing Shortcut, File, and Folder Issues .....	63
Other Issues .....	65
User Account Issues .....	66
Limits .....	67
Document History .....	68

# What is Amazon WAM

---

Amazon WorkSpaces Application Manager (Amazon WAM) offers a fast, flexible, and secure way for you to deploy and manage applications for [Amazon WorkSpaces](#). Amazon WAM accelerates software deployment, updates, patching, and retirement by packaging Microsoft Windows desktop applications into virtual containers that run as though they are installed natively.

Amazon WAM is fully integrated with the AWS Management Console, and allows you to build an application catalog from your line-of-business applications, third-party applications that you own the license for, and applications purchased through the AWS Marketplace. Simple, pay-as-you-go, user-based pricing ensures that you only pay for the applications that are used.

Amazon WAM is available in two versions: Amazon WAM Lite and Amazon WAM Standard. With Amazon WAM Lite, you can manage and deliver applications from the AWS Marketplace free of charge. You pay only for the applications that your users activate. With Amazon WAM Standard, you can build your application catalog with line-of-business applications, third-party applications for which you own licenses, and applications from the AWS Marketplace for Desktop Apps. You also gain access to additional features.

## Available Regions

Amazon WAM is available in the following regions.

Code	Name
us-east-1	US East (N. Virginia)
us-west-2	US West (Oregon)
eu-west-1	EU (Ireland)
ap-southeast-1	Asia Pacific (Singapore)
ap-southeast-2	Asia Pacific (Sydney)

## Related Services

Amazon WAM is only available on [Amazon WorkSpaces](#). For more information, see the [Amazon WorkSpaces FAQ](#).

# Setting Up Amazon WAM

---

To use Amazon WorkSpaces Application Manager (Amazon WAM), you must meet the following prerequisites.

## Topics

- [AWS Account \(p. 2\)](#)
- [Amazon WorkSpaces \(p. 2\)](#)
- [Controlling Access to Amazon WAM Resources \(p. 3\)](#)

## AWS Account

Your AWS account gives you access to all services, but you are charged only for the resources that you use.

If you do not have an AWS account, use the following procedure to create one.

### To sign up for AWS

1. Open <http://aws.amazon.com/> and choose **Create an AWS Account**.
2. Follow the online instructions.

Your root account credentials identify you to services in AWS and grant you unlimited use of your AWS resources, such as your WorkSpaces. To allow other users to manage AWS Directory Service resources without sharing your security credentials, use AWS Identity and Access Management (IAM). We recommend that everyone work as an IAM user, even the account owner. You should create an IAM user for yourself, give that IAM user administrative privileges, and use it for all your work.

## Amazon WorkSpaces

Amazon WAM requires that you have an Amazon WorkSpaces directory and users. Your users access the Amazon WAM applications that are assigned to them on Amazon WorkSpaces, so any user that you assign an application to must also have a Workspace provisioned for them. For more information, see the [Amazon WorkSpaces Administration Guide](#).

# Controlling Access to Amazon WAM Resources

Amazon WAM must have permission to perform certain actions on your behalf. You can grant this access using IAM roles.

By default, IAM users don't have permission to access Amazon WAM resources. To allow an IAM user to perform actions on Amazon WAM resources, you must create a policy that grants the user permission to access Amazon WAM.

## Contents

- [Create the Application Packaging Role \(p. 3\)](#)
- [Create the AWS Marketplace Access Role \(p. 3\)](#)
- [\(Optional\) Grant an IAM User Access to Amazon WAM \(p. 4\)](#)

## Create the Application Packaging Role

This IAM role allows the Amazon WAM packaging instance to access your application package catalog. If you have not already done so, create the `AmazonWamAppPackaging` role using the following steps.

### To create an IAM role to access your Amazon WAM application catalog

1. Open the Identity and Access Management (IAM) console at <https://console.aws.amazon.com/iam/>.
2. In the navigation pane, choose **Roles** and then choose **Create New Role**.
3. On the **Set Role Name** page, enter `AmazonWamAppPackaging` as the name of the role and then choose **Next Step**.

#### Important

If you do not specify `AmazonWamAppPackaging` as the name of the role, the packaging and validation applications can't access your packages.

4. On the **Select Role Type** page, choose **AWS Service Roles**, and then choose **Select** next to **Amazon EC2**.
5. On the **Attach Policy** page, select the **AmazonWorkSpacesApplicationManagerAdminAccess** policy and then choose **Next Step**.
6. On the **Review** page, choose **Create Role**.

## Create the AWS Marketplace Access Role

This IAM role allows Amazon WAM to access the AWS Marketplace on your behalf. The first time you log in to the Amazon WAM console, you are prompted to create a role with the name `AmazonWamMarketplace_Default_Role`. You must allow this role to be created.

The following is the IAM policy for the `AmazonWamMarketplace_Default_Role` role.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "catalog-admin:Describe*",
        "catalog-admin:Get*"
      ]
    }
  ]
}
```



```
        "catalog-admin:Search*",
        "catalog-admin:List*",
        "catalog-admin:CreateListing",
        "catalog-admin:UpdateListing",
        "catalog-admin>DeleteListing",
        "catalog-user:SimulateView*",
        "catalog-user:SimulateGet*",
        "catalog-user:SimulateBrowse*"
    ],
    "Resource": "*"
  }
]
```

## (Optional) Grant an IAM User Access to Amazon WAM

The following IAM policy allows an IAM user or group of users to administer Amazon WAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "catalog-admin:*",

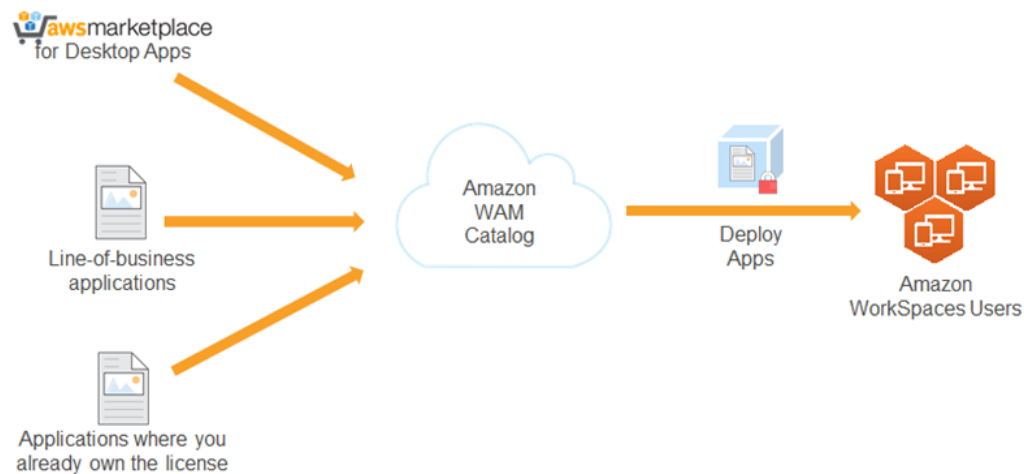
        "ds:*",
        "iam:ListAttachedRolePolicies",
        "iam:ListRoles",
        "iam:CreateRole",
        "iam:PutRolePolicy"
      ],
      "Resource": "*"
    }
  ]
}
```

For more information about embedding a policy in a user or group, see [Working with Inline Policies](#) in the *IAM User Guide*.

# Getting Started with Amazon WAM

---

Using Amazon WorkSpaces Application Manager (Amazon WAM), you can deploy subscriptions to your Amazon WorkSpaces users from the AWS Marketplace, your line-of-business applications, or applications where you already own the licenses. The following illustration shows the process to deploy applications:



The following topics explain how to get started using Amazon WAM.

## Topics

- [Sign Up for Amazon WAM \(p. 5\)](#)
- [Build Your Application Catalog \(p. 6\)](#)
- [Assign Applications to Your Users \(p. 8\)](#)
- [Amazon WorkSpaces Application Manager Client Application \(p. 8\)](#)

## Sign Up for Amazon WAM

Amazon WAM is available to all Amazon WorkSpaces customers. After you finish setting up Amazon WorkSpaces, you can start using Amazon WAM console by going to the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/> and then choosing **Application catalog** in the navigation bar. The first time you open the Amazon WAM console, two Amazon EC2 AMIs are shared with your account:

#### Amazon WAM Studio AMI

This AMI contains the Amazon WorkSpaces Application Manager Studio application to package your own applications.

#### Amazon WAM Admin Player AMI

This AMI contains the Amazon WAM Admin Player application to validate your application packages.

Amazon WAM will publish new AMIs when they become available. You will automatically receive the latest AMIs to replace your older version.

## Build Your Application Catalog

You can build your application catalog from three sources:

- The AWS Marketplace
- Your line-of-business applications
- Applications for which you already own the license

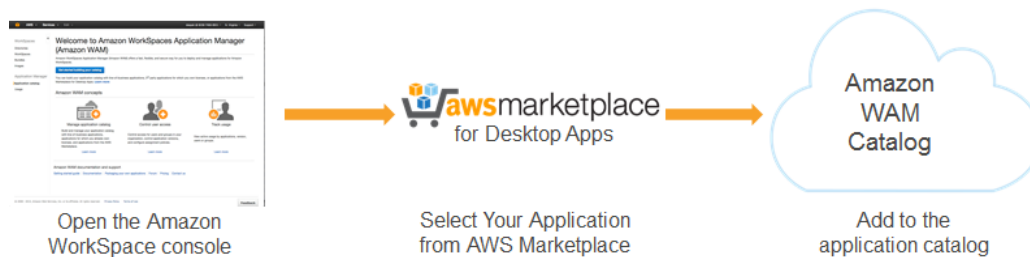
For more information about how to get started building your application catalog, see [Building Your Application Catalog \(p. 9\)](#).

#### Topics

- [Add an AWS Marketplace Application to Your Application Catalog \(p. 6\)](#)
- [Add Your Application to Your Application Catalog \(p. 7\)](#)

## Add an AWS Marketplace Application to Your Application Catalog

The following graphic shows the process of adding an AWS Marketplace application to your application catalog.

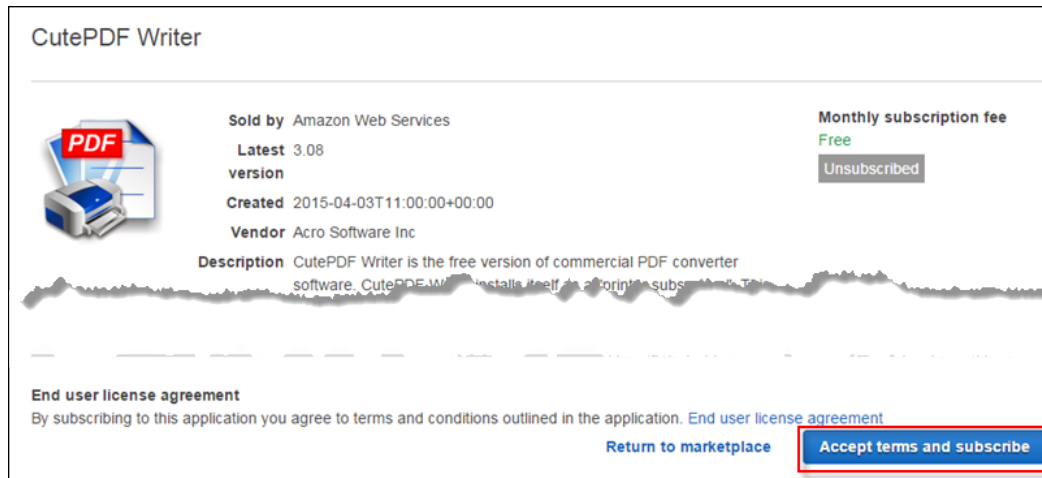


### To add an AWS Marketplace application to your application catalog

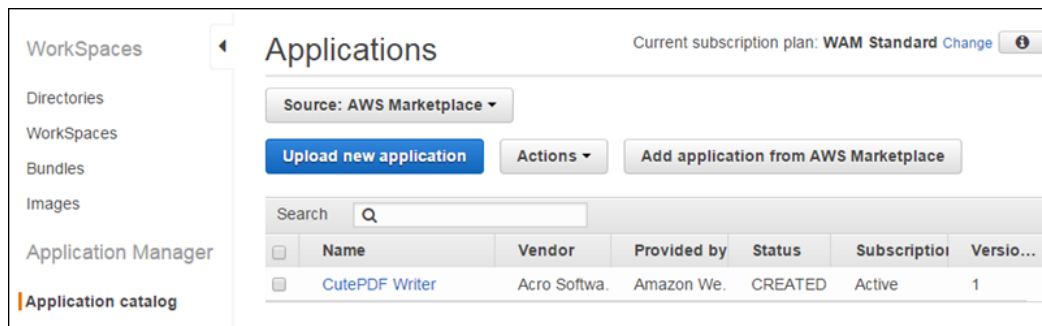
This procedure uses CutePDF Writer as the AWS Marketplace application to add to your application catalog. You can use this procedure to add any AWS Marketplace application to your application catalog.

1. Open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/> and choose **Applications** in the navigation pane.
2. In **Search**, type **CutePDF writer** and press Enter.

3. Choose **CutePDF Writer**. In the CutePDF Writer detail page, choose **Accept terms and subscribe**.



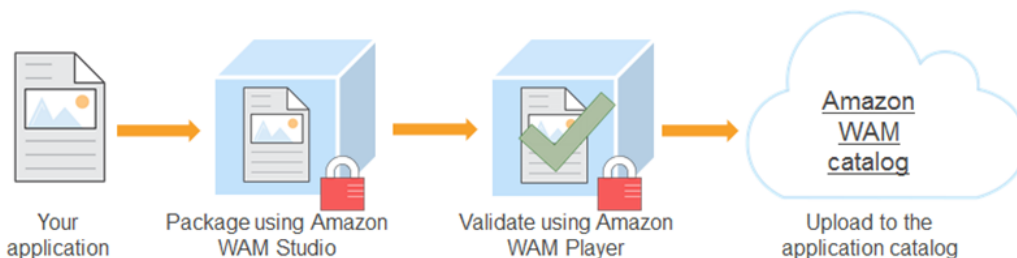
4. In **Applications**, set **Source** to **AWS Marketplace**. **CutePDF Writer** appears in the list of subscribed applications.



After you add an AWS Marketplace application to your application catalog, you can add your own application to the application catalog or assign the AWS Marketplace application to your users. To add your own application, see the next procedure. To assign your application to your users, see [Assign Applications to Your Users](#) (p. 8).

## Add Your Application to Your Application Catalog

The following graphic shows the process of adding to the application catalog your line-of-business applications and applications for which you already own the license.



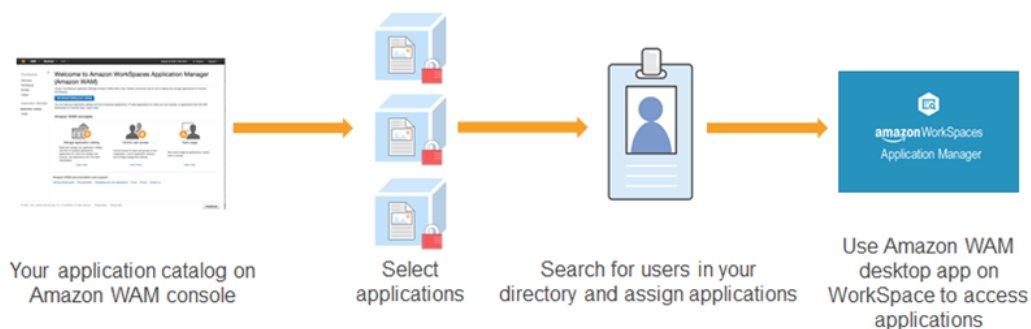
To add your application to your application catalog, see the following:

- [Packaging an Application with Amazon WorkSpaces Application Manager Studio \(p. 20\)](#)
- [Validating an Application Package with Amazon WAM \(p. 35\)](#)
- [Uploading Your Own Applications \(p. 10\)](#)

After you add your application to your application catalog, you can assign your application to your users. For more information, see [Assign Applications to Your Users \(p. 8\)](#).

## Assign Applications to Your Users

Your users only have access to the Amazon WAM applications that you assign to them. The following is the process to assign an application in your application catalog to a user.



For more information about assigning applications to your users, see [Assigning an Application to Users and Groups \(p. 13\)](#).

## Amazon WorkSpaces Application Manager Client Application

Your users use the Amazon WorkSpaces Application Manager client application on their WorkSpace to access the applications that you assign to them. The installer is available on your user's WorkSpace desktop. If the installer is missing from your user's WorkSpace, it will be added the next time the user reboots the WorkSpace. Your users must install this application to gain access to the applications that you assign to them. For more information, see the [Amazon WAM User Guide](#).

# Managing Your Amazon WAM Applications

---

The following topics describe how to manage your Amazon WorkSpaces Application Manager (Amazon WAM) applications.

## Topics

- [Building Your Application Catalog \(p. 9\)](#)
- [Assigning an Application to Users and Groups \(p. 13\)](#)
- [Updating an Application \(p. 14\)](#)
- [Monitoring Application Usage \(p. 17\)](#)
- [Removing All Application Assignments \(p. 17\)](#)
- [Troubleshooting Amazon WAM Application Issues \(p. 18\)](#)

## Building Your Application Catalog

All of your Amazon WAM applications are listed in your application catalog, which contains all of the applications that you can assign to your users. This includes both applications that you subscribe to from the AWS Marketplace, as well as applications that you upload.

## Topics

- [Subscribing to an Application From AWS Marketplace \(p. 9\)](#)
- [Uploading Your Own Applications \(p. 10\)](#)
- [Removing an Application From Your Application Catalog \(p. 11\)](#)

## Subscribing to an Application From AWS Marketplace

In addition to your own application, you can subscribe to applications from the AWS Marketplace with a monthly subscription.

### To subscribe to an application from the AWS Marketplace

1. Open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/> and choose **Applications** in the navigation pane.

2. Choose **Add application from AWS Marketplace**.
3. Select the application to add to your catalog.
4. In the application detail page, select the version or versions of the application to add to your catalog and choose **Accept terms and subscribe**.
5. In the application list page, choose **Return to my application catalog**.
6. If needed, set **Source** to **AWS Marketplace**. The application is added to the list of subscribed applications.

## Uploading Your Own Applications

To make your own applications available in your application catalog, you create a new application from an application package. For more information about creating an application package, see [Packaging an Application with Amazon WorkSpaces Application Manager Studio \(p. 20\)](#).

### To create a new application from an application package

Before beginning this procedure, you must have at least one validated application package.

1. Open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/> and choose **Applications** in the navigation pane.
2. Choose **Upload new application** and perform the following steps:
3. In the **Enter application details** page, enter the following fields and choose **Next**.

#### Package

Enter the name of the application.

#### Short description

Enter a short description of the application.

#### Description

Enter a description of the application.

#### Category

Select the appropriate application category.

#### Provided by

Enter the name to display as the provider of the application.

#### Vendor

Enter the name to display as the vendor of the application.

4. In the **Enter support details** page, enter the following fields and choose **Next**. All of these fields are optional.

#### Email contact

Enter the email address users should use to obtain support for the application.

#### Support link

Enter the URL that users should use to obtain support for the application.

#### Support description

Enter any additional support information.

5. In the **Select application package** page, enter the following fields and choose **Review**.

#### Package details

##### Package title

Select the title of the package to create the application from. The list contains all of your validated application packages.

##### Package build

Select the package build for the application.

#### Version information

##### Version title

Enter a short name for the version.

##### Description

Enter the description of the version.

#### Application licensing

##### License type

Select the license type for the application.

##### By WorkSpace

The license is per WorkSpace.

##### By user

The license is per user.

#### Application constraints

##### Maximum install

Enter the maximum number of installations allowed for the application.

6. In the **Review** page, review the application information, make any necessary changes, and choose **Confirm and upload**. The application may not be immediately available.

## Removing an Application From Your Application Catalog

You may want to remove an application from Amazon WorkSpaces because you want to deprecate an existing application for a different application or your user has stopped using an existing application. You can remove any application from AWS Marketplace application or an uploaded application from your application catalog.

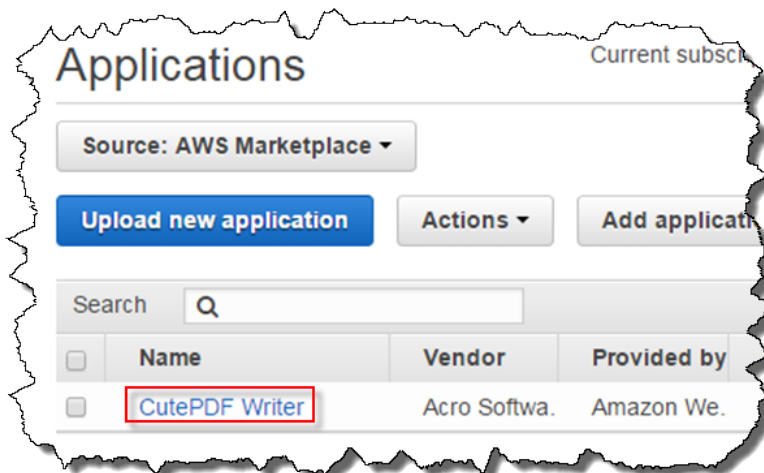
#### Important

Before you can remove an application, you must remove all assignments for the application. For more information, see [Users and Groups \(p. 16\)](#).

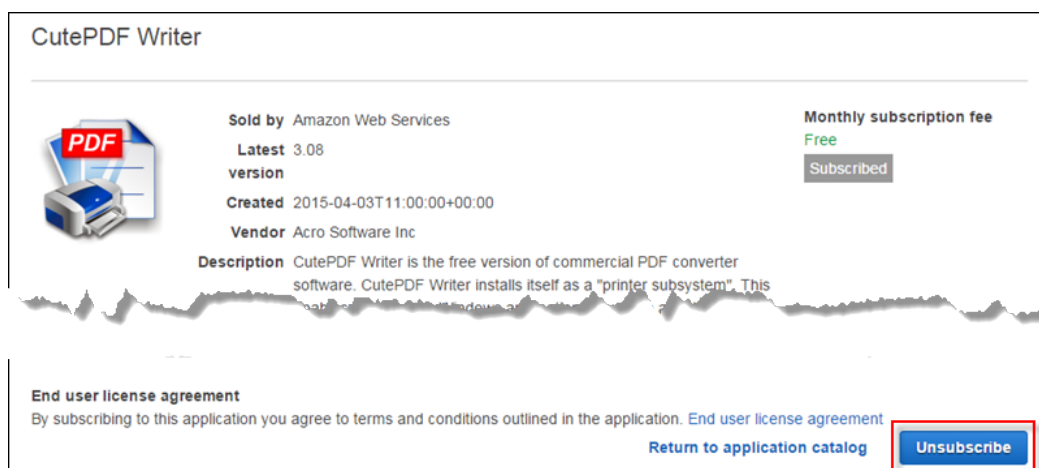
#### To remove an AWS Marketplace application from your application catalog

1. Open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/> and choose **Applications** in the navigation pane.
2. Set **Source** to **AWS Marketplace** and choose the application to remove from your application catalog. The link opens the application product page.



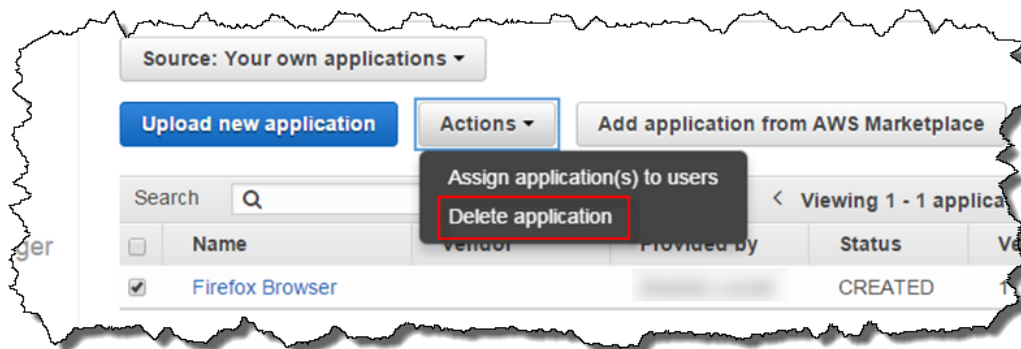


3. In the application product page, choose **Unsubscribe**.



### To remove an application you uploaded from your application catalog

1. Open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/> and choose **Applications** in the navigation pane.
2. Set **Source** to **Your own applications**.
3. Choose the check box for the application you want to remove from your application catalog.
4. Choose **Actions** and select **Delete application**.



## Assigning an Application to Users and Groups

To assign an application to users or groups, perform the following steps:

### Note

Application assignments by user groups only work up to one level of group hierarchy. Only the users who are direct members of the group will receive the assigned applications. Users who are members of sub-groups within the assigned group will not receive the assigned applications.

1. Open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/> and choose **Applications** in the navigation pane.
2. If the application is an uploaded application, set **Source** to **Your own applications**. If the application is a subscribed application, change **Source** to **AWS Marketplace**.

### Note

You will only be able to assign your uploaded applications on Amazon WAM Standard.

3. Select up to five applications to assign, choose **Actions**, and select **Assign application(s) to users**.
4. In the **Select users** page, make the following selections and choose **Search**.

### Directory

Select the AWS Directory Service directory in which to search for users or groups.

### Type

Select the type of entity for which to search, users or groups.

### Fields containing

Enter all or part of the user or group name for which to search. Leave this field blank to search for all users or groups.

If there are more search results than can be displayed in one page, you can scroll through the pages by clicking on the left arrow (<) or right arrow (>) in the list header.

5. From the **Search results** list, select the user or group to which to assign the application, and choose the right arrow (>) between the two lists. The selected items are moved to the **Selected** list.

To remove an item from the **Selected** list, select the item and choose the left arrow (<) between the two lists. The selected items are moved to the **Search results** list.

You can change the search criteria at any time to add other users or groups to the **Selected** list. The items in the **Selected** list are not removed when you redo the search.

Repeat this step until all of the users and groups to which to assign the application are selected. When you are ready, choose **Next**.

6. In the **Configure assignment options** page, make the following selections, and choose **Review**.

**Note**

These selections are only available in Amazon WAM Standard. Amazon WAM Lite has limited selections.

**Version**

Select the application version to assign.

**Installation Type**

Select one of the following:

**Optional**

The application will be installed when the user initiates the installation from the Amazon WorkSpaces Application Manager client application.

**Required**

The application will automatically be installed on the user's Workspace.

**Auto update**

Select one of the following:

**No**

You need to assign application updates manually to your users.

**Yes**

The application is updated automatically when a new version of the application is published.

7. In the **Review** page, review the assignments, make any changes needed, and choose **Confirm and assign**. The selected applications are assigned to the selected users and group.

## Updating an Application

From the application detail page, you can perform the following tasks:

- View and edit the application versions, and create a new version
- View and edit the constraints for the application
- View and edit the tags for the application
- View and edit the assignments for the application
- View the support details for the application

Amazon WAM Lite customers will have limited features. Amazon WAM Standard customers will be able to assign users to the applications you uploaded and configure the assignment options such as controlling versions, managing the update settings, or configuring the installation type.

### To open the detail page for an application

1. Open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/> and choose **Applications** in the navigation pane.
2. To view the details for one of your own applications, set **Source** to **Your own applications**. To view the details for a subscribed application, change **Source** to **AWS Marketplace**.
3. Open the application detail page by choosing the name of the application.
4. If the application is one of your own applications, you can edit the information about the application, such as the name, vendor, provider, and description. To edit these, choose the pencil icon next to the application name, enter the new information, and choose **Save**.

#### Topics

- [Versions \(p. 15\)](#)
- [Constraints \(p. 16\)](#)
- [Tags \(p. 16\)](#)
- [Users and Groups \(p. 16\)](#)
- [Support Details \(p. 16\)](#)

## Versions

You can view the version information for all of the applications in your catalog, regardless of the source. You can also update the versions for your own applications. To view the versions for an application, in the application detail page, expand the **Versions** section.

If the application is one of your own applications, you can perform the following tasks:

- [Create a New Version \(p. 15\)](#)
- [Edit an Existing Version \(p. 15\)](#)

Amazon WAM Lite subscribers will not be able assign any versions of the uploaded applications non any versions of the applications from the AWS Marketplace for Desktop Apps that are not the latest version.

## Create a New Version

You can create a new version of an application. Before you can create a new version of the application, you must have uploaded a new version of the application package as explained in [Creating a New Version of an Application \(p. 27\)](#).

### To create a new version of an application

1. In the **Versions** section of the application details page, choose **Create new version**.
2. In the **Select application package** page, enter the following fields and choose **Submit**. The new application version is created.

#### Package details

##### Package build

Select the package build for the new version.

#### Version information

##### Version title

Enter a short name for the new version.

##### Description

Enter the description of the new version.

## Edit an Existing Version

You can modify the title and description of an existing version.

### To edit an existing version of an application

1. In the **Versions** section of the application details page, select the version to edit, and choose **Edit version**.
2. Modify the version title and description as required and choose **Submit**. It takes several minutes for the updated version information to be available.

## Constraints

You can view and modify the constraints for an application from the application details page.

### To view the constraints for an application

1. In the application detail page, expand the **Constraints** section. The application constraints are displayed.
2. Here, you can perform the following tasks:
  - To add a constraint, choose **Add constraints**.
  - To create a new version of the application, choose **Create new version**, and perform the procedure in [Assigning an Application to Users and Groups \(p. 13\)](#).
  - To edit an existing version of the application, select the version to edit and choose **Edit version**. You can modify the version title and description.

## Tags

Amazon WAM Standard customers can view, add, and remove tags from your applications.

### To view the tags for an application

1. In the application detail page, expand the **Tags** section. The application tags are displayed.
2. Here, you can perform the following tasks:
  - To add a tag, enter the tag name and value, and choose **Add tag**.
  - To remove a tag, select the tag and choose **Delete tag**.

## Users and Groups

You can view and modify the users and groups to which your applications are assigned. Amazon WAM Lite customers will be unable to assign users and groups to your uploaded applications. Amazon WAM Standard customers will be able to assign users and groups to your uploaded applications.

### To view the users and groups to which an application is assigned

1. In the application detail page, expand the **Users and Groups** section. The users to which the application is assigned are displayed.
2. Here, you can perform the following tasks:
  - To assign the application to another user or group, choose **Add user or group**, and perform the procedure in [Assigning an Application to Users and Groups \(p. 13\)](#).
  - To update the assignment options for the application, choose **Edit assignment options**. On the [Configure assignment options \(p. 14\)](#) page, you can update the assignment options.
  - To remove an application assignment, select the users and groups to remove the assignment from, choose **Remove user or group**, verify that you want to perform the action, and choose **Continue**. The application is no longer assigned to the selected users and groups.

## Support Details

To view the support information for an application, in the application detail page, expand the **Support details** section. The support information for the application is displayed.

## Monitoring Application Usage

Amazon WAM allows you to track your application usage by viewing usage reports in the Amazon WAM console.

Topics

- [Applications \(p. 17\)](#)
- [Users \(p. 17\)](#)

### Applications

To view application usage information, perform the following steps.

#### To view application usage information

1. Open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/> and choose **Usage** in the navigation pane.
2. In the **Usage** page, choose the **Application** tab. All of your applications are displayed in the list.

If there are more search results than can be displayed in one page, you can scroll through the pages by choosing the left arrow (<) or right arrow (>) in the list header.

You can view additional information about a specific application by choosing the expander button next to the product.

### Users

To view usage information by user, perform the following steps.

#### To view usage information by user

1. Open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/> and choose **Usage** in the navigation pane.
2. In the **Usage** page, choose the **Users** tab.
3. Under **Directory**, choose your directory. All of your users that have assigned applications are displayed in the list.

#### Note

Users are not displayed in the list until they have connected to their WorkSpace after the Amazon WorkSpaces Application Manager client application has been installed.

If there are more search results than can be displayed in one page, you can scroll through the pages by choosing the left arrow (<) or right arrow (>) in the list header.

You can view the applications that are assigned to a specific user by choosing the expander button next to the username.

## Removing All Application Assignments

The Amazon WAM service automatically registers itself with an AWS Directory Service directory. This allows Amazon WAM to access the directory and prevents the directory from being deleted while it is in use by Amazon WAM. Before you can delete the directory that your WorkSpaces are assigned to, you

must remove all Amazon WAM application assignments in the directory and deregister Amazon WAM from the directory. To make this task easier, the Amazon WAM console provides the ability to remove all application assignments and deregister itself from a directory in a single operation.

**Note**

Even if you do not have any application assignments, you must perform the following procedure to deregister Amazon WAM from a directory.

**To remove all application assignments and deregister Amazon WAM from a directory**

1. Open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/> and choose **Usage** in the navigation pane.
2. In the **Usage** page, choose the **Users** tab.
3. Under **Directory**, choose the directory. All of the directory users that have assigned applications are displayed in the list.

**Note**

Users are not displayed in the list until they have connected to their WorkSpace after the Amazon WorkSpaces Application Manager client application has been installed.

4. Choose **Remove all assignments**. The process to remove all application assignments and deregister the directory is initiated immediately. The entire process takes 10 to 15 minutes to complete. After all applications assignments have been removed, the service is deregistered from the directory, which is removed from the list of directories in the **Users** tab.

## Troubleshooting Amazon WAM Application Issues

The following are possible issues that you might have with Amazon WAM applications.

Issues

- [I assigned an application to a user, but I don't see the user in the usage report \(p. 18\)](#)
- [I don't know how to manage User Access Control on Windows \(p. 18\)](#)
- [I am performing HTTPS Proxy Inspection on my WorkSpaces \(p. 19\)](#)

### I assigned an application to a user, but I don't see the user in the usage report

You won't see the user in the usage report until the user has launched the Amazon WorkSpaces Application Manager client application for the first time. For more information, see [Users \(p. 17\)](#).

### I don't know how to manage User Access Control on Windows

For apps that trigger UAC input, the behavior with Amazon WAM is similar to natively installed apps. Amazon WAM does not introduce any changes. You can manage UAC through the same policies that you regularly use for managing user permissions.

For apps that use a script as part the of application installation (for example, KMS activation), the script can be set to run as local system administrator. If the user on a WorkSpace is not a local administrator, Amazon WAM requires the following two registry settings to be set to allow the script to execute within the package:

- HKLM\SOFTWARE\Amazon\StreamingCore\Settings\AppEvent\AllowLocalSystem,DWORD,"1"

- HKLM\SOFTWARE\Amazon\StreamingCore\Settings\Driver\PhysicalLayerMode,DWORD,"1"

## I am performing HTTPS Proxy Inspection on my WorkSpaces

All Amazon WAM endpoints are accessed over port 443 (HTTPS). If you are routing WorkSpace outbound traffic using a corporate proxy and performing HTTPS content inspection, we recommend that you exclude the Amazon WAM endpoints to prevent validation or latency issues.

For Amazon WAM to function correctly, the following endpoints must be accessible over HTTPS:

- wam-idb.*region-code*.amazonaws.com
- wam-ps.*region-code*.amazonaws.com
- s3.amazonaws.com
- \*.s3.amazonaws.com
- s3-external-1.amazonaws.com
- \*.s3-external-1.amazonaws.com

For a list of regions that support Amazon WAM, see [Available Regions \(p. 1\)](#).



# Packaging and Validating Your Applications

---

To create Amazon WorkSpaces Application Manager (Amazon WAM) applications of your own making, you must create the application package, and validate that the package installs and works correctly. This is accomplished using two special EC2 instances.

## Topics

- [Packaging an Application with Amazon WorkSpaces Application Manager Studio \(p. 20\)](#)
- [Validating an Application Package with Amazon WAM \(p. 35\)](#)
- [Updating an Existing Package \(p. 37\)](#)
- [Finding a Package with Amazon WAM \(p. 56\)](#)
- [Sharing a Package with Amazon WAM \(p. 56\)](#)
- [Deleting a Package with Amazon WAM \(p. 57\)](#)
- [Troubleshooting Amazon WAM Packaging Issues \(p. 57\)](#)

## Packaging an Application with Amazon WorkSpaces Application Manager Studio

An application package is created on a special EC2 instance. The first time you access the Amazon WAM console, an Amazon EC2 AMI is shared with you. Use this AMI to launch the packaging instance.

## Topics

- [Best Practices for Application Packaging \(p. 21\)](#)
- [Launching the Packaging Instance \(p. 21\)](#)
- [Creating a New Application Package \(p. 21\)](#)
- [Creating a New Version of an Application \(p. 27\)](#)
- [Understanding Color Coding \(p. 27\)](#)
- [Packaging Specific Applications \(p. 30\)](#)
- [Using Licenses in Your Package \(p. 34\)](#)

## Best Practices for Application Packaging

The following are the recommended best practices to use when packaging an application:

- Always launch a new packaging instance when creating a new application package.
- For best performance, we recommend using an m3.xlarge or better instance type for the packaging instance. Normal Amazon EC2 instance charges apply.
- Always review the virtualization settings after capturing the application installation. For more information, see [Installation Capture Information \(p. 24\)](#).

## Launching the Packaging Instance

We recommend that you launch the packaging and validation instances into their own VPC to make launching and configuration easier. For this, you need the following:

- A VPC in a region that Amazon WAM is supported in.
- A public subnet in the VPC.

Before launching the packaging instance, you must create the `AmazonWamAppPackaging` IAM role. For more information, see [Create the Application Packaging Role \(p. 3\)](#).

To launch a packaging instance, follow the steps in the [Launching Your Instance from an AMI](#) topic in the *Amazon EC2 User Guide for Windows Instances*. The following are specific settings for launching a packaging instance:

- In step 2, select the region in which your VPC is located.
- In step 4, select the **My AMIs** tab, select **Shared with me**, and select the **Amazon WAM Studio AMI**.
- In step 5, for best performance, we recommend **m3.xlarge** or better for the instance type. You will be charged for AWS resources such as the EC2 instance, Amazon S3 storage, and bandwidth when using this instance to package your applications.
- In step 6, on the **Configure Instance Details** page, enter the following:
  - Under **Network**, select your VPC.
  - Under **Subnet**, select the public subnet in the VPC.
  - Under **IAM role**, select the **AmazonWamAppPackaging** role, which you created in [Create the Application Packaging Role \(p. 3\)](#).
- In step 9, on the **Configure Security Group** page, select a security group that allows access to the instance over port 3389 (RDP) from your network.

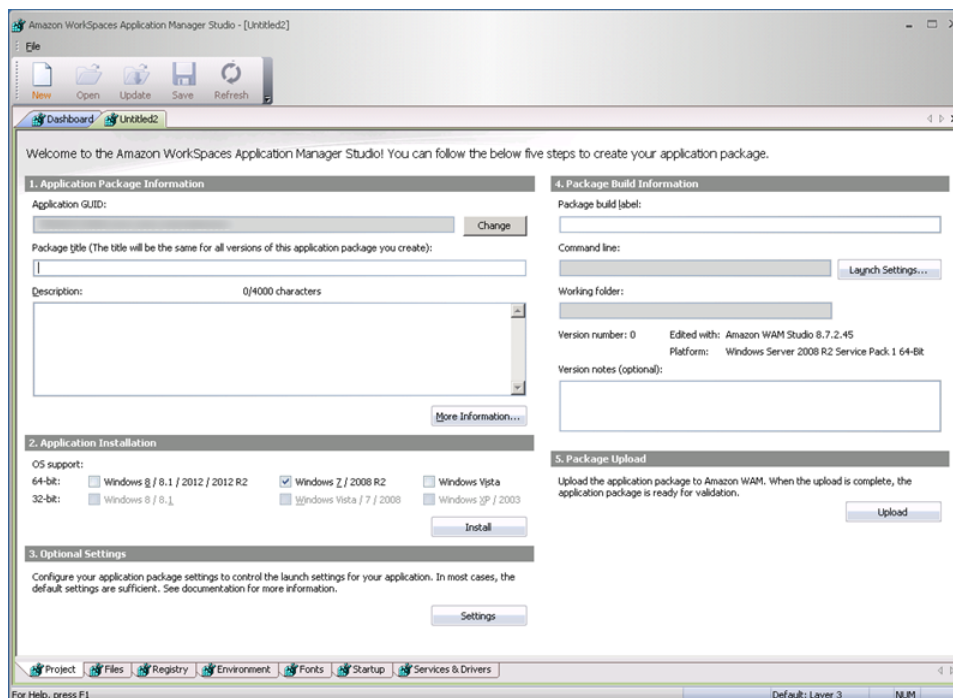
## Creating a New Application Package

After the packaging instance is launched, use it to create an application package. AWS recommends that you launch an entirely new packaging instance for each application package that is created.

### To package an application

1. Using a Remote Desktop client, connect to the packaging instance. The remaining instructions are performed on the packaging instance.
2. Download the installer for the application to be packaged to the instance. All Amazon WAM applications run on Amazon WorkSpaces. If the application to be packaged is available for different platforms, you should download the installer for Windows.

3. Launch the Amazon WorkSpaces Application Manager Studio by opening the desktop shortcut.



4. In the Amazon WorkSpaces Application Manager Studio toolbar, click **New** to create a new application project. Fill in information as noted below.

To package an application, enter the following information:

- [Application settings \(p. 22\)](#)
- [Application Installation \(p. 22\)](#)
- [Optional Settings \(p. 25\)](#)
- [Package Build Information \(p. 26\)](#)
- [Package Upload \(p. 26\)](#)

## Application settings

Enter information in the following fields:

### Application GUID

The identifier of the application. This is used by Amazon WAM to identify a unique application package for troubleshooting and debugging. An identifier is automatically created when you create a new application project. Choose **Change** to generate a new identifier if needed.

### Package title

The name of the application.

### Description

A description of the application.

## Application Installation

Select the target operating system for the application and capture the application installation by performing the steps in [Capturing the Application Installation \(p. 23\)](#).

## Capturing the Application Installation

When creating an application package, you install the application on the packaging instance. During the installation, the Amazon WorkSpaces Application Manager Studio application captures all changes to the file system, registry, environment, and other systems during the installation.

Before capturing the application installation, save the application project. This is required before you can capture the application installation because some applications require that the system be restarted to complete.

The installation process is different for [applications that have an installer \(p. 23\)](#) and [applications that do not have an installer \(p. 24\)](#).

### Topics

- [Applications With an Installer \(p. 23\)](#)
- [Applications Without an Installer \(p. 24\)](#)
- [Captured Information \(p. 24\)](#)

### Applications With an Installer

This procedure is used for applications that have an installer.

#### To capture the installation of an application that has an installer

1. Click **Install**.
2. Enter the following fields in the **Capture Application Installation** dialog box:

##### **Installer path**

Enter the path of the application installer.

##### **Command line parameters**

Enter any command line parameters needed for the installation.

##### **Compatibility Settings**

Click **Compatibility Settings** and enter any Windows compatibility settings that are required by the installer, such as being run as administrator.

##### **Capture the file and registry activity for the following processes**

###### **Installer process and sub-processes only (default)**

Only changes made by the installer process, and any processes the installer creates, are captured.

###### **All processes**

Changes made by all processes are captured.

##### **Ignore changes under the Installer path specified above**

File changes made to the folder that contains the installer are not captured. This is useful if the installation process creates temporary files in the installer directory that are not required by the installed application.

##### **After capturing is complete, do the following actions during the merge:**

###### **Replace short path names with long ones in the registry**

Any short (8.3 format) path names captured in the registry are replaced with long path names, if possible.

3. Click **Launch** and install the application as needed. All changes to the files system, registry, and environment are captured during the installation. For more information about what is captured, and how to modify that information, see [Captured Information \(p. 24\)](#).
4. When the installation is complete, click **OK** to close the **Capture** dialog box.

## Applications Without an Installer

This procedure is used for applications that do not have an installer, such as applications that are delivered as standalone executables, or are delivered in an archive, such as a ZIP file.

### To capture the installation of an application that does not have an installer

1. Click **Install**.
2. Enter the following fields in the **Capture Application Installation** dialog box:

#### Installer path

Enter `explorer.exe`.

#### Command line parameters

Leave this field empty.

#### Compatibility Settings

Do not modify these settings.

#### Capture the file and registry activity for the following processes

Select **All processes**.

#### Ignore changes under the Installer path specified above

Leave this set to the default setting.

#### After capturing is complete, do the following actions during the merge:

##### Replace short path names with long ones in the registry

Leave this set to the default setting.

3. Click **Launch**. A Windows Explorer window is opened.
4. Perform all steps needed to install the application, such as copying the standalone executable to the desired location, or extracting the files from a ZIP archive. All changes to the files system are captured during the installation. For more information about what is captured, and how to modify that information, see [Captured Information](#) (p. 24).
5. Close the Windows Explorer window and click **OK** to close the **Capture** dialog box.

## Captured Information

The tabs at the bottom of the Amazon WorkSpaces Application Manager Studio contain all of the information that was captured during the installation of the application. Review each of these and modify them as needed for your package. Care must be taken when modifying these as making changes can make the application inoperable.



### Installation Capture Information

#### Files

All changes to the file system are contained here. You can add, remove, or move files to customize your package.

#### Registry

All changes to the system registry are contained here. You can add, remove, or move registry entries to customize your package.

#### Environment

Any changes to the environment variables are contained here. You can add, remove, or modify these variables to customize your package.

#### Fonts

All changes to the system fonts are contained here. You can add or remove fonts to customize your package.

### **Startup**

All changes to the system startup folders and registry keys are contained here. You can add, remove, or modify these to customize your package.

### **Services & Drivers**

All services and drivers that were installed during the installation are contained here. You can add or remove these to customize your package.

## Optional Settings

Choose **Settings** and enter any packager settings that are required. In most cases, the default settings are used.

### **Note**

Except for the **Virtualization** field, you can configure these settings by creating a new version of the package. The virtualization setting can only be set on a new package.

### **Configurable AppEvents**

Add any additional app events needed for the package. These include any actions you want to take place after certain application events, such as application launch or registration. For more information, see [Adding Required Applications or Scripts \(p. 50\)](#).

### **Virtualization**

Add settings and options that affect how the application is virtualized. An application is virtualized when it is installed on the client.

#### **Environment Variables**

Specify if the package environment variables are available to the entire system, or just to the application.

#### **Fonts**

Specify if the package fonts are available to the entire system, or just to the application.

#### **Startup Items**

Specify if the startup folder and registry keys are automatically run when the application is virtualized.

#### **Special Requirements**

Specify additional actions needed when the application is virtualized.

### **Sandboxing**

Choose how the application is sandboxed on the client. For more information, see [Sandboxing Your Package \(p. 53\)](#).

#### **Folder Exclusions**

Add any folders to exclude from the package.

#### **Registry Key Exclusions**

Add any registry keys to exclude from the package.

#### **Disposition for new registry key roots**

Specifies what access the local system has to the application's resources, such as files, registry settings, and so on.

##### **Virtual-integrated (layer 3)**

Resources can be seen by both the application and the local system, but are not physically installed on the local system. This is the default setting.

##### **Virtual-isolated (layer 4)**

Resources can only be seen by the application, and are not physically installed on the local system.

### **Application settings**

Enter additional settings for the application.

#### **Preserve application settings when application is removed**

Specifies if the application-specific settings modified by the user should be preserved or removed when the application assignment is removed from the user. If this field is set

to preserve these settings, the application settings will be available if the application is assigned to the user again.

**Allow Windows to roam application settings**

Specifies if the application-specific settings should be written to the standard Windows roaming profiles location.

**Security**

Add or modify how processes on the client can access the application's virtual resources. For more information, see [Setting File Security \(p. 55\)](#).

## Package Build Information

Enter the following fields.

**Package build label**

Enter a label for the package. This is displayed in the Amazon WAM console as the version label.

**Command line**

Choose **Launch Settings** and enter the command to launch the installed application, along with any other launch and compatibility settings that are needed for the application.

**Important**

This is the command that is used to launch the application on the client, not the application installer.

**To set the command used to launch the application on the client**

1. On the **Files** tab, find the application's main executable file in the list of files.
2. Right click on the file and choose **Set As Command Line**. The application command line and working folder are automatically populated to point to the selected file.

**Working folder**

Set this to the working folder for the application.

All packages must have a command line. If the package you are creating does not have a primary application, you should set this to `explorer.exe`.

After you upload a package, you cannot change the command line for the package. If you need to modify the command line, you need to create an entirely new package.

## Package Upload

Upload the package for validation by performing the following steps:

**To upload the package for validation**

1. Save the application project.
2. Choose **Upload** to upload the application package to the Amazon WAM service. When the upload is complete, choose **Close** in the **Progress** dialog box.
3. Close the application project. Do not modify a project after it has been uploaded. If you need to update a package, create a new version of the package. For more information, see [Creating a New Version of an Application \(p. 27\)](#).

The application package is now ready for validation. For more information, see [Validating an Application Package with Amazon WAM \(p. 35\)](#).

## Creating a New Version of an Application

In addition to creating new application packages, you can also create new versions of existing packages. This is particularly useful for distributing patches and updates for your applications. AWS recommends that you launch an entirely new packaging instance for each application package version that is created. To launch a packaging instance, follow the steps in [Launching the Packaging Instance \(p. 21\)](#).

After the packaging instance is launched, use it to create a new application package version. If an application patch is being created, you can reuse a packaging instance that only has the application being patched installed on it.

## Creating an Application Update

An application update (or patch) is an update to an existing application that requires that the application already be installed. AWS recommends using this approach to apply security updates, application updates, and minor version upgrades.

### To create an application update

1. Using a Remote Desktop client, connect to the packaging instance. The remaining instructions are performed on the packaging instance.
2. Download the update installer or new application version to the instance.
3. Launch the Amazon WorkSpaces Application Manager Studio by opening the desktop shortcut.
4. In the Amazon WorkSpaces Application Manager Studio **Dashboard**, select the application to create an update for.
5. Choose **Update** on the Amazon WorkSpaces Application Manager Studio toolbar. If the application is already installed in the instance, you will see a message box that the files on the instance will be overwritten by the files in the package. Choose **Yes**. The message box displays a list of files that were overwritten.

#### Important

If you double-click the package or choose **Open**, the package opens in read-only mode. You cannot update the package in this mode.

6. Capture the upgrade installation as described in [Capturing the Application Installation \(p. 23\)](#).
7. If you need to modify any of the optional settings, choose **Settings** and modify any packager settings that are required. In most cases, the default settings are used. For more information, see [Optional Settings \(p. 25\)](#).
8. In the **Package build label** field of the **Package Build Information** section, enter the label for the new version. In the **Version notes (optional)** field, enter a description for the new version.

#### Note

You cannot change the command line for a package that has been uploaded. If you need to modify the command line, you need to create an entirely new package.

9. Upload the new package version as described in [Package Upload \(p. 26\)](#).

The new version of the application package is now ready for validation. The new version cannot be assigned to users until the validation is completed. For more information, see [Validating an Application Package with Amazon WAM \(p. 35\)](#).

## Understanding Color Coding

You can determine the status of an item in the dashboard by looking at the color. If the item has a conflict, hover over the item to see a message.

The following table shows the meaning of each color.



### Item codes

Color / attribute	Item type	Description
Black	All items	Normal.
Grey	Template folders	The folder does not contain any files.
Grey Italic	Files Folders and subfolders Registry keys and values	The item is excluded from the package.
Blue	Files Folders and subfolders Registry keys and values	The item was added after the package was created. This color does not apply to items that were manually added to the package.
Magenta	Files Folders and subfolders Registry keys and values	The item was changed after the package was created. This color does not apply to items that were manually added to the package.
Red	Files Folders and subfolders Registry keys and values	The item was skipped during package creation.
Red	All items	The item is missing from the package or the item has a conflict. For more information, see the following table on conflict codes.
Bold	Folder	Working folder.
Bold	File	Command line.

Red text in the tabs represent a conflict in your package. The following sections describe the conflicts in the tabs.

#### Files tab

Field	Meaning
Source File	The file for the item is not in the package. Add the missing file to the package.

#### Fonts tab

Field	Meaning
Action	The fonts are set to be registered, but the font will not be visible to the operating system

Field	Meaning
	because the Fonts key is set to Virtual-isolated (layer 4).
<b>Action and File Disposition</b>	The fonts are set to be registered, but the font will not be visible to the operating system because the font file is set to Virtual-isolated (layer 4).
<b>File Path</b>	The file for the item is not in the package. Add the missing file to the package.

### Startup tab

Field	Meaning
<b>Name and Command Disposition</b>	The Registry Run (or RunOnce) key asset cannot be started by the system because the key is set to Installed-permanent (layer 1) or Installed-temporary (layer 2), but its target command line is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4) so it will not be visible to the system.
<b>Command Disposition</b> for entries under <b>HKCU Run</b> or <b>HKCU RunOnce</b>	One of these two conditions exists: <ul style="list-style-type: none"> <li>• The shortcut file does not exist on the system.</li> <li>• The startup shortcut cannot be started by the operating system because it is set to Installed-permanent (layer 1) or Installed-temporary (layer 2), but its target command line is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4) so it will not be visible to the operating system.</li> </ul>

### Services & Drivers tab

Field	Meaning
<b>Action</b>	This action cannot be performed on a service whose type is not specified.
<b>Error Control</b>	The error control type is not specified.
<b>Interactive</b>	The interactive setting is not specified.
<b>Image File</b>	The driver is set to start on boot or system start, but its service key or image file is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4) so it will not be visible to the system at that time.
<b>Image File and Key Disposition</b>	The image file layer is greater than the service key layer, which can yield inconsistent behavior except if the service key is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4).

Field	Meaning
<b>Name</b>	The service is set to restart or reboot automatically upon failure. This will cause the service to always restart when the application is shut down, which will put it back into the Running state. To fix this issue, set the failure action of the service to <b>Turn Failure Actions OFF</b> .
<b>Start</b>	The start type is not specified.
<b>Type</b>	The service type is not specified.

## Packaging Specific Applications

This section shows you how to package specific applications.

### Topics

- [Packaging 7-Zip with Amazon WAM \(p. 30\)](#)
- [Packaging Google Chrome with Amazon WAM \(p. 31\)](#)
- [Packaging Mozilla Firefox with Amazon WAM \(p. 33\)](#)

## Packaging 7-Zip with Amazon WAM

To create a package, launch a packaging instance and a validating instance on Amazon EC2 and then create and validate a new package. For more information, see the following topics:

- [Packaging an Application with Amazon WorkSpaces Application Manager Studio \(p. 20\)](#)
- [Packaging and Validating Your Applications \(p. 20\)](#)

### To create a package

To create a new package, use Windows Remote Desktop to connect to your package instance and then use Amazon WorkSpaces Application Manager Studio to create the package. For more information, see [Connecting to Your Windows Instance Using RDP](#).

1. Go to the [7-Zip download page](#) and download the Windows installer for 7-Zip.
2. Double-click the desktop shortcut to start the Amazon WorkSpaces Application Manager Studio.
3. On the Amazon WorkSpaces Application Manager Studio dashboard, choose **New**.
4. Under **1. Application settings**, do the following:
  - For **Title**, enter the name of the application. This text appears in the **Dashboard** tab and in the Amazon WorkSpaces Application Manager Studio, to identify this application.
  - For **Description**, enter a short description for the application.
5. In **2. Application Installation**, select an operating system and choose **Install**.
6. In the **Capture Application Installation** dialog box, do the following:
  - For **Installer path**, enter the path to the 7-Zip installer.
  - For **Capture file and registry activity for the following processes**, select **Installer process and sub-processes only**.
  - Select **Ignore changes under the Installer path specified above**.

- For **After capturing is complete, do the following actions during the merge**, select **Replace short path names with long ones in the registry**.
- Choose **Launch**.

The installer for 7-Zip starts after you choose **Launch**. Follow the steps in the installer wizard to complete the installation.

7. Choose the **Files** tab and do the following:
  - In the **<ProgramFiles64>7-Zip** directory, open the context (right-click) menu for **7-zip.dll** and choose **Properties**.
  - In the **File Properties** dialog box, select **Installed temporary (layer 2)** and then choose **OK**.
8. To ensure that Windows Explorer does not hold a reference to `7-zip.dll` when the application is removed from the WorkSpace, create a configurable AppEvent that deregisters the file by using a batch file.

To create the batch file, do the following:

- On the packaging instance desktop, start Notepad.
- Type the following and save the file to the desktop as `unreg_7zip.bat`.

```
@echo off
regsvr32 /s /u "c:\Program Files\7-Zip\7-zip.dll"
```

To create the configurable AppEvent, do the following:

- In Amazon WorkSpaces Application Manager Studio dashboard, under **3. Optional settings**, choose **Settings**.
  - In the **Settings** dialog box, choose **Configurable AppEvents**.
  - In the **Configurable AppEvent Setting** dialog box, do the following:
    - Under **Trigger**, choose **Before deactivation**.
    - In **File name** under **Handler command line**, choose the `unreg_7zip.bat` file you created earlier.
    - Under **Handler launching**, select **Run elevated (only supported on Windows Vista or later)**.
    - Under **After launching**, select **Wait for the handler to exit**. In time out after, type 5 and select **minute(s)**.
    - Choose **OK**.
  - Choose **OK**.
9. Skip **3. Optional settings**.
  10. Choose **Save** in the toolbar to save the application package project.
  11. Select the **Project** tab. In **4. Package Build**, do the following:
  12. In **5. Package upload**, choose **Upload** and then **Close**.

## Packaging Google Chrome with Amazon WAM

To create a package, launch a packaging instance and a validating instance on Amazon EC2 and then create and validate a new package. For more information, see the following topics:

- [Packaging an Application with Amazon WorkSpaces Application Manager Studio \(p. 20\)](#)
- [Packaging and Validating Your Applications \(p. 20\)](#)

## To create a package

To create a new package, use Windows Remote Desktop to connect to your package instance and then use Amazon WorkSpaces Application Manager Studio to create the package. For more information, see [Connecting to Your Windows Instance Using RDP](#).

1. In the Amazon WorkSpaces Application Manager Studio toolbar, choose **New**.
2. In **1. Application settings**, do the following:
  - In the **Title** field, enter the name of the application. This text appears on the **Dashboard** tab and in the Amazon WorkSpaces Application Manager Studio to identify this application.
  - In the **Description** field, enter a short description for the application.
3. In **2. Application installation**, select an operating system and choose **Install**.
4. In the **Capture Application Installation** dialog box, do the following:
  - In the **Installer path** field, type `C:\Program Files (x86)\Internet Explorer\iexplore.exe`.
  - In the **Command line parameter** field, type the URL to the Windows installer for Chrome browser at [Alternate \(offline\) Google Chrome installer \(Windows\)](#).
  - For **Capture file and registry activity for the following processes**, select **Installer process and sub-processes only**.
  - Select **Ignore changes under the Installer path specified above**.
  - For **After capturing is complete, do the following actions during the merge**, select **Replace short path names with long ones in the registry**.
  - Choose **Launch** and follow the instructions to install Chrome offline on Windows for all accounts.
  - After installing Chrome, choose **OK** and **Close**.
5. Skip **3. Optional settings**.
6. Choose the **Files** tab, open the context (right-click) menu for `<ProgramFilesX86>\Google\Chrome\Application\chrome.exe`, and choose **Properties**. In the **Properties** dialog box, select **Installed-temporary (layer 2)** under **Disposition** and choose **OK**.
7. Choose the **Registry** tab and then do the following:
  - Open the context (right-click) menu for `HKEY_LOCAL_MACHINE\Software` and choose **Add Key**.
  - In the **Add Registry Key** dialog box, select **Create from this existing registry key**, and choose **Browse**.
  - In the **Browse System Registry** dialog box, choose `HKEY_LOCAL_MACHINE\Software\Policies` and choose **OK** twice.
  - Open the context (right-click) menu for `HKEY_LOCAL_MACHINE\Software\Policies`, and choose **Add Key**.
  - In the **Add Registry Key** dialog box, select **Create a new key with the following name**, type `Google`, and choose **OK**.
  - Repeat the previous two steps to create the `HKEY_LOCAL_MACHINE\Software\Policies\Google\Update` registry key.
  - Open the context (right-click) menu for `HKEY_LOCAL_MACHINE\Software\Policies\Google\Update`, and choose **Add Value**.
  - For **Name** in the **Add Registry Value** dialog box, type `AutoUpdateCheckPeriodMinutes`. For **Type**, select **REG\_DWORD**. For **Data**, type `0`, and choose **OK**.
8. Choose the **Project** tab and then do the following in **4. Package build**:
  - For **Label**, enter a label for the package. This label appears in the Amazon WAM console as the version label.

- For **Command line**, choose **Launch Settings**.
  - In the **Command line** field of the **Application Launch Settings** dialog box, type "start the Chrome browser from the Desktop shortcut, the Start menu, or the Quick Launch toolbar." Select **Set the working folder to the folder of the command line executable, if any**, select **Set the icon to the source file of the command line executable, if any**, and choose **OK**.
9. Choose **Save** on the toolbar to save the application package project.
  10. In **5. Package upload**, choose **Upload** and **Close**.

## Packaging Mozilla Firefox with Amazon WAM

To create a package, launch a packaging instance and a validating instance on Amazon EC2 and then create and validate a new package. For more information, see the following topics:

- [Packaging an Application with Amazon WorkSpaces Application Manager Studio \(p. 20\)](#)
- [Packaging and Validating Your Applications \(p. 20\)](#)

### Important

This procedure shows how to disable automatic updates. You will need to manually update the version. To create a version update, see [Updating an Application \(p. 14\)](#).

### To create a package

To create a new package, use Windows Remote Desktop to connect to your package instance and then use Amazon WorkSpaces Application Manager Studio to create the package. For more information, see [Connecting to Your Windows Instance Using RDP](#).

1. Start Internet Explorer and download the Windows installer for Firefox Extended Support Release at [Download Firefox Extended Support Release in your language](#).
2. Double-click the desktop shortcut to start Amazon WorkSpaces Application Manager Studio.
3. On the Amazon WorkSpaces Application Manager Studio Dashboard, choose **New**.
4. In **1. Application settings**, do the following:
  - For **Title**, enter the name of the application. This text appears on the **Dashboard** tab and in the Amazon WorkSpaces Application Manager Studio to identify this application.
  - For **Description**, enter a short description for the application.
5. In **2. Application installation**, select an operating system and choose **Install**.
6. In the **Capture Application Installation** dialog box, do the following:
  - For **Installer path**, enter the path to the Firefox installer.
  - For **Command line parameter**, type `-ms` to do a silent install.
  - For **Capture file and registry activity for the following processes**, select **Installer process and sub-processes only**.
  - Select **Ignore changes under the Installer path specified above**.
  - For **After capturing is complete, do the following actions during the merge**, select **Replace short path names with long ones in the registry**.
  - Choose **Launch**.

The installer for Firefox starts after you choose **Launch**.

7. On the desktop of the packaging instance, start Notepad, and then do the following to create a configuration file that disables the auto-update feature and sets the homepage and a script file to use the configuration file:

- Copy the following code into Notepad and save the file as C:\Program Files (x86)\Mozilla Firefox\mozilla.cfg:

```
// Disable automatic updates
pref("app.update.enabled", false);
pref("app.update.auto", false );
pref("app.update.mode", 0);
pref("app.update.service.enabled", false);

// Disable the default browser check
pref("browser.shell.checkDefaultBrowser", false);
pref("browser.startup.homepage_override.mstone", "ignore");

// Set the homepage
pref("browser.startup.homepage", "https://www.amazon.com");
```

- Copy the following code into Notepad and save the file as C:\Program Files (x86)\Mozilla Firefox\defaults\pref\local-settings.js:

```
//
pref("general.config.obscure_value", 0);
pref("general.config.filename", "mozilla.cfg");
```

#### Note

The first line in the configuration and files must be two forward slashes (//).

8. Choose the **Files** tab, and then do the following: Open the context (right-click) menu for **<ProgramFilesX86>\Mozilla Firefox** and then choose **Add File**. In **Source file name** on the Add File dialog box, enter **C:\Program Files (x86)\Mozilla Firefox\mozilla.cfg**, and then choose **OK**.
  - Open the context (right-click) menu for **<ProgramFilesX86>\Mozilla Firefox** and then choose **Add File**. In **Source file name** on the Add File dialog box, enter **C:\Program Files (x86)\Mozilla Firefox\mozilla.cfg**, and then choose **OK**.
  - Open the context (right-click) menu for **<ProgramFilesX86>\Mozilla Firefox\defaults\pref** and then choose **Add File**. In **Source file name** on the Add File dialog box, enter **C:\Program Files (x86)\Mozilla Firefox\defaults\pref\local-settings.js**, and then choose **OK**.
9. Choose the **Project** tab and then do the following in **4. Package build**:
  - For **Label**, enter a label for the package. This label appears in the Amazon WAM console as the version label.
  - For **Command line**, choose **Launch Settings**. In the **Application Launch Settings** dialog box, enter **C:\Program Files (x86)\Mozilla Firefox\firefox.exe**. Select **Set the working folder to the folder of the command line executable, if any**, select **Set the icon to the source file of the command line executable, if any**, and then choose **OK**.
10. Choose **Save** on the toolbar to save the application package project.
11. In **5. Package upload**, choose **Upload** and then **Close**.

## Using Licenses in Your Package

How you add an application license depends on the type of license.

Topics

- [Using a Single License on All Instances \(p. 35\)](#)
- [Using an ISV License Service \(p. 35\)](#)
- [Using a License per Instance \(p. 35\)](#)

## Using a Single License on All Instances

For an application that uses a single master license for all users, enter the license at the appropriate screen when you package the application. The license is captured in the package and used in all instances.

## Using an ISV License Service

For an application using an ISV license service hosted at a website, enter the service URL at the appropriate screen when you package the application. The information provided by the service to activate the application is captured in the package and used in all instances.

You can also automatically connect to the license service by adding an environment variable with the license service location to each WorkSpaces client and then use a Configurable AppEvent to run a script to read the environment variable to get to the license service to activate the application.

### To use a Configurable AppEvent to connect to an ISV license service

This procedure assumes that your WorkSpaces clients have an environment variable with license service location and you have a script that use the license service to activate the application.

#### Note

You cannot add a Configurable AppEvent to a new version of a package.

1. Copy your script to the package instance.
2. Under **3. Optional settings** in Amazon WorkSpaces Application Manager Studio, choose **Settings**.
3. In the **Settings** dialog box, choose **Configurable AppEvents** and **Add**.
4. In the **Configurable AppEvent Settings** dialog box, do the following:
  - For **Trigger**, choose **After launch**.
  - For **Frequency**, choose **Fire the first time only**.
  - For **Handler**, in the **File name** field, enter the name of the script to activate the application.
  - Choose **OK** twice to close the dialog boxes.

## Using a License per Instance

For an application using one license per instance, enter the license information at the appropriate screen when the application installs on the computer.

# Validating an Application Package with Amazon WAM

An application package is validated on a special EC2 instance. The first time you access the Amazon WorkSpaces Application Manager console, an Amazon EC2 AMI is shared with you. Use this AMI to launch the validation instance.



## Topics

- [Launching the Validation Instance \(p. 36\)](#)
- [Validating the Application Package \(p. 36\)](#)

# Launching the Validation Instance

We recommend that you launch the validation instance into its own VPC to make launching and configuration easier. You can, and should, use the same VPC that you used for the packaging instance. For more information, see [Launching the Packaging Instance \(p. 21\)](#).

Before launching the validation instance, you must create the `AmazonWamAppPackaging` IAM role. For more information, see [Create the Application Packaging Role \(p. 3\)](#).

To launch a validation instance, follow the steps in the [Launching Your Instance from an AMI](#) topic in the *Amazon EC2 User Guide for Windows Instances*. The following are specific settings for launching a validation instance:

- In step 2, select the region in which your VPC is located.
- In step 4, select the **My AMIs** tab, and select **Shared with me**, and select the **Amazon WAM Admin Player AMI**.
- In step 5, for best performance, we recommend **m3.xlarge** or better for the instance type. Normal Amazon EC2 instance charges apply.
- In step 6, on the **Configure Instance Details** page, enter the following:
  - Under **Network**, select your VPC.
  - Under **Subnet**, select the public subnet in the VPC.
  - Under **IAM role**, select the **AmazonWamAppPackaging** role, which you created in [Create the Application Packaging Role \(p. 3\)](#).
- In step 9, on the **Configure Security Group** page, select a security group that allows access to the instance over port 3389 (RDP) from your network.

# Validating the Application Package

After the validation instance is launched, use it to validate your pending application packages. AWS recommends that you launch an entirely new validation instance for each application package that is created.

## To validate an application package

1. Using a Remote Desktop client, connect to the validation instance. The remaining instructions are performed on the validation instance.
2. Launch the Amazon WAM Admin Player and select **Pending Apps** in the navigation bar. Any applications that need to be validated are displayed.
3. Select the application package to be validated. If the package is a new package, click **Install**. If the package is a new version of an existing package that is already installed on the instance, click **Upgrade**. The application is installed or upgraded on the validation instance.
4. Launch the application as your users will, and verify that it works correctly.

If the application package does not install or run as expected, you can update the package by creating a package update and validating the update. For more information, see [Creating an Application Update \(p. 27\)](#).

5. Click **Approve** to mark the package as testing completed.

Using the application package, you can now create a new application, or create a new version of an existing application. For more information about creating a new application, see [Uploading Your Own Applications \(p. 10\)](#). For more information about creating a new version of an existing application, see [Create a New Version \(p. 15\)](#).

## Updating an Existing Package

After you create an application package, you can modify the contents of the package according to the needs of your users. For example, you may want to include additional files that your users need, change registry entries to work in your environment, or set environment variables that are unique to your users.

This section shows you how to update the contents of your application package. To update the package content, you first have to open the package in update mode.

### To update the package and create a new version

1. Connect to your packaging instance and launch Amazon WorkSpaces Application Manager Studio. The remaining instructions are performed on the package instance.
2. In the **Dashboard** tab, choose your application package, and then choose **Update** on the toolbar.
3. Make your changes to the package.
4. In the **Label** field of **4. Package build** section, enter a label for the updated version.
5. Save the updated package.
6. Upload the new package version as described in [Package Upload \(p. 26\)](#).

After you update the package, you have to validate the package before your users can download the new version.

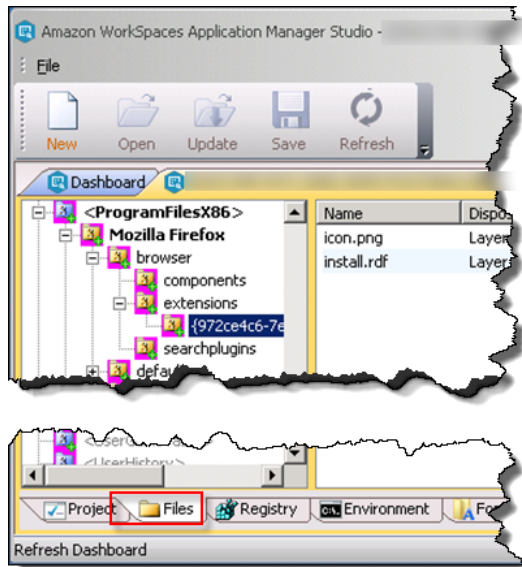
### Topics

- [Working with Files and Folders \(p. 37\)](#)
- [Working With the Registry \(p. 41\)](#)
- [Working with Environment Variables \(p. 43\)](#)
- [Working with Fonts \(p. 44\)](#)
- [Working with Startup Items \(p. 46\)](#)
- [Working with Services and Drivers \(p. 48\)](#)
- [Adding Required Applications or Scripts \(p. 50\)](#)
- [Sandboxing Your Package \(p. 53\)](#)
- [Setting File Security \(p. 55\)](#)

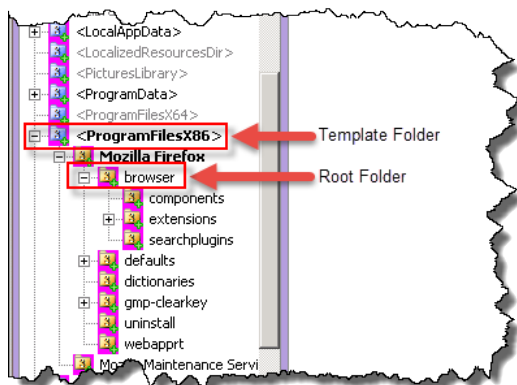
## Working with Files and Folders

You can change the files and folders in your application package to suit the needs of your users. For example, you may want all of your users to have a folder that contains specific files. Or you may want to reduce the size of your package by removing unneeded folders and files.

You can change the folders and files in your package by choosing the **Files** tab in the project dashboard.



Your package contains two kinds of folders:



Template folders map to the actual system folder paths in the package instance. Template folders are in blue and cannot be removed or added to the package.

Root folders map to the same path shown in the navigation pane. These folders are in yellow and were added during package creation or manually. You can add root folders to any part of the folder tree.

### To add a root folder or subfolder

1. In the **Files** tab, open the context (right-click) menu in the left navigation bar and choose **Add Root Folder**. To add a subfolder, open the context (right-click) menu for a folder in the left navigation bar and choose **Add Subfolder**.
2. In the **Add Folder** dialog box, do one of the following:
  - To create a new root folder, choose **Create new root folder with the following name** and enter a path for it.
  - To create a new root folder from an existing folder in the package instance, choose **Create from existing source** and then choose **Browse** to select the folder to add to the package. However, the files and subfolders in the root folder are not added to the package. To add all of the subfolders in the root folder, choose **Recursively include subdirectories**. To add the files

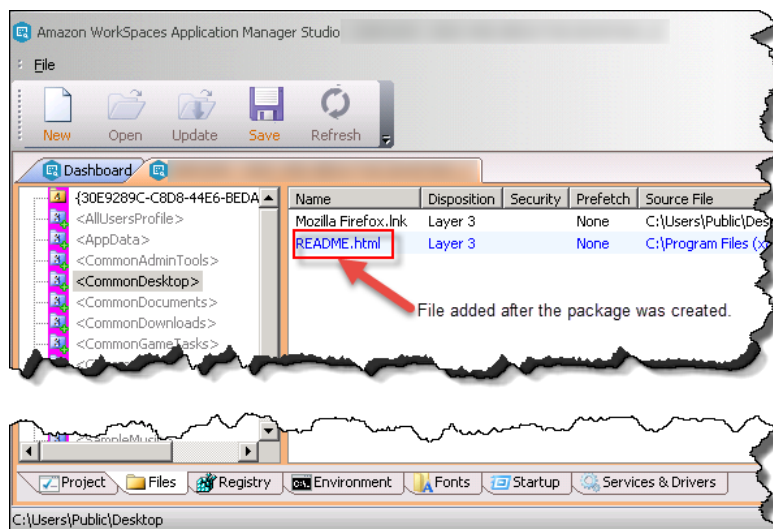
in the folder and subfolders, choose **Include files within the folder(s)**. To use a different name for the folder, choose **Override existing folder name with the following name** and type the new name for the folder.

3. Choose **OK**.

### To add a file to a folder

1. If the file is not in the package instance, add the file to the instance.
2. In the **Files** tab, open the context (right-click) menu for the folder where you want to add the file and choose **Add File**.
3. In the **Add File** dialog box, choose **Browse** and then choose the file to add to the package.
4. To change the name of the file, choose **Override source file name with the following name** and enter the new name of the file.
5. Choose **OK**.

Folders and files added after the package was created appear in magenta.



### To rename a root folder or file

1. In the **Files** tab, open the context (right-click) menu for the root folder or file that you want to rename and choose **Rename**.
2. Type the new name of the root folder or file and press **Enter**.

### To edit the properties of a root folder or file

1. In the **Files** tab, open the context (right-click) menu for the root folder or file that you want to rename and choose **Properties**.
2. In the dialog box, change the properties of the folder and choose **OK**.

### To set a folder as the working folder

The working folder contains the application executable or some related files. Some applications only work if the working folder is set to a folder that does not contain the application executable file. For these types of applications, you have to set the working folder to another folder.

1. In the **Files** tab, open the context (right-click) menu for the root folder that you want to be the working folder and choose **Set As Working Folder**.
2. In the dialog box, change the properties of the folder and choose **OK**.

### To remove a folder or file from the package

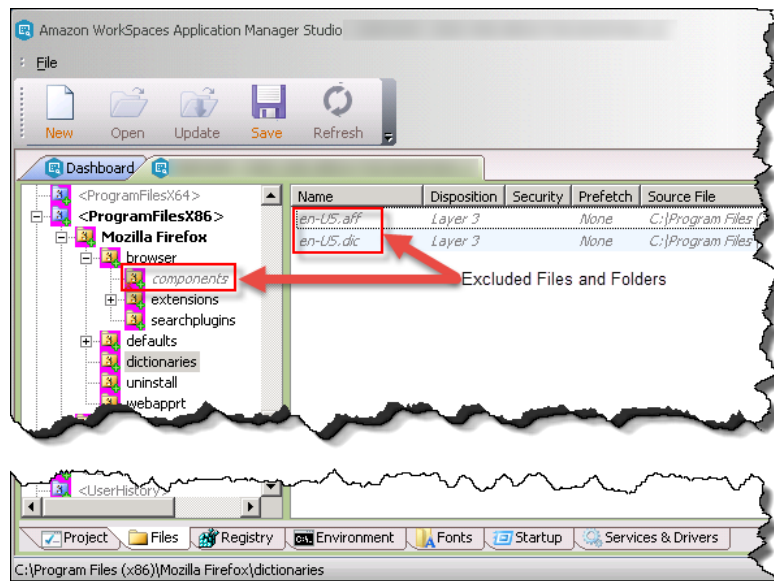
- In the **Files** tab, open the context (right-click) menu for the folder or file that you want to remove and then choose **Delete**.

### To exclude a folder or file from the package

Folders, subfolders, and files that are excluded in the package are not installed in Amazon WorkSpaces, but are still part of the package. This gives you the flexibility of including certain folders and files for one package deployment and not for another. Excluding folders and files also enables you to test an package without a folder or file before permanently deleting it from the project.

- In the **Files** tab, open the context (right-click) menu for the root folder or file that you want to exclude from the package and then choose **Exclude**.

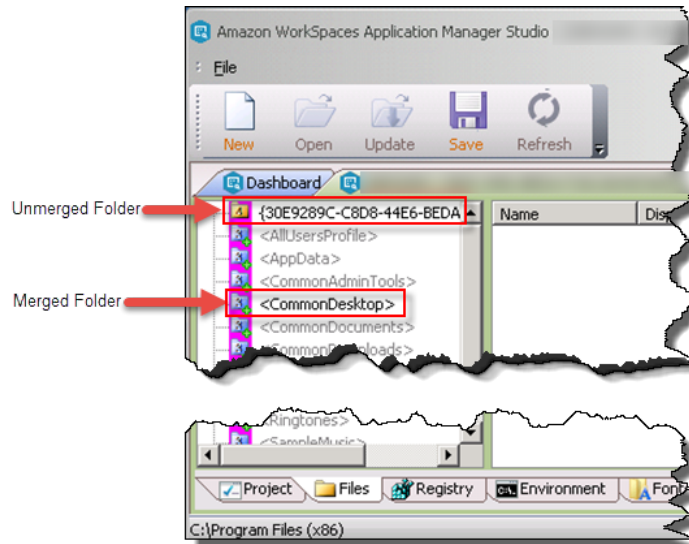
Excluded folders and files appear in gray italicized text.



### To change an unmerged folder into a merged folder

If your package has a root folder that is already on the Amazon WAM client application, both package and client application folders are merged into a single folder. If you are assigning several packages that use the same folder, the contents of each folder are added to a single merged folder.

Merged folders have a plus sign (+).



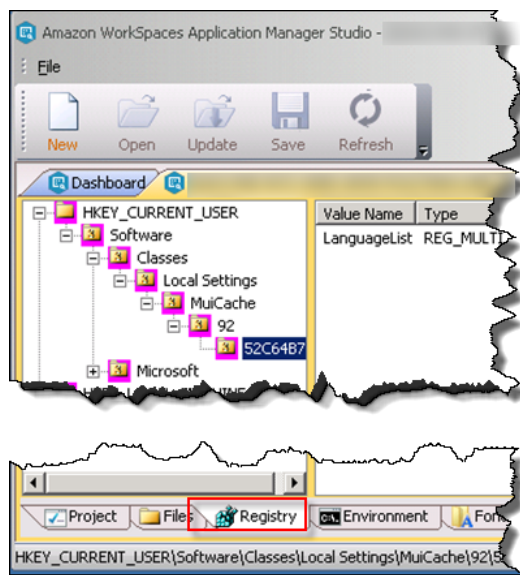
- In the **Files** tab, open the context (right-click) menu for the unmerged folder, and then choose **Merged Folder**.

#### To show a folder or file in Explorer

- In the **Files** tab, open the context (right-click) menu for the folder or file that you want to show in Explorer and then choose **Show in Explorer**.

## Working With the Registry

You can modify the registry in your package by choosing the **Registry** tab in the project dashboard. You can create a new subkey or add an existing subkey under any key, including root keys (or hives). However, you cannot create new root keys.



### To add a subkey

1. In the **Registry** tab, open the context (right-click) menu for the root or subkey where you want to add a subkey and then choose **Add Key**.
2. In the **Add Registry Key** dialog box, create a new key by entering a new name or using an existing key to create a new key.

If you create a new key from an existing key, you can choose one or more of the following options:

#### **Recursively include subkeys**

Add all subkeys under the existing key. Clear this choice if you only want to add the selected key.

#### **Include values within the key(s)**

Add the values of the keys. Clear this choice to add only the key.

#### **Use existing registry key instead of current key as addition target**

Adds the complete path from the top level of the new key to save you from having to manually re-create the entire path, starting with the top level.

3. Choose **OK**.

### To rename a registry key

1. In the **Registry** tab, open the context (right-click) menu for the root or subkey for which you want to change the name, and then choose **Rename**.
2. Change the name of the key and press Enter.

### To change the disposition of a registry key

1. In the **Registry** tab, open the context (right-click) menu for the root or subkey for which you want to change the name, and then choose **Disposition**. Choose **Apply setting to subkeys recursively** to apply the same disposition to the subkeys.
2. Choose **OK**.

### To add a value to the key

1. In the **Registry** tab, open the context (right-click) menu for the root or subkey for which you want to change the value, and then choose **Add Value**.
2. In the **Add Registry Value** dialog box, do the following:
  - For **Name**, enter the name of the registry value.
  - For **Type**, select the data type of the value.
  - For **Data**, enter the registry value. Choose **Display as hexadecimal** to display the hexadecimal value.

#### **Note**

If the registry value appears to contain an environment variable, such as %ServerName%, Amazon Studio attempts to resolve the environment variable. If the environment variable does not exist, then Amazon Studio escapes the % characters. %ServerName% becomes %%ServerName%%. To use an environment variable as the value, define the environment variable in the **Environment** tab.

3. Choose **OK**.

### To edit a value

1. In the **Registry** tab, open the context (right-click) menu for the root or subkey for which you want to change the value, and then choose **Edit Value**.
2. In the **Edit Registry Value** dialog box, make your changes.
3. Choose **OK**.

### To delete a key or value

1. In the **Registry** tab, open the context (right-click) menu for the root or subkey for which you want to change the value, and then choose **Delete**.
2. Choose **OK**.

### To exclude a key

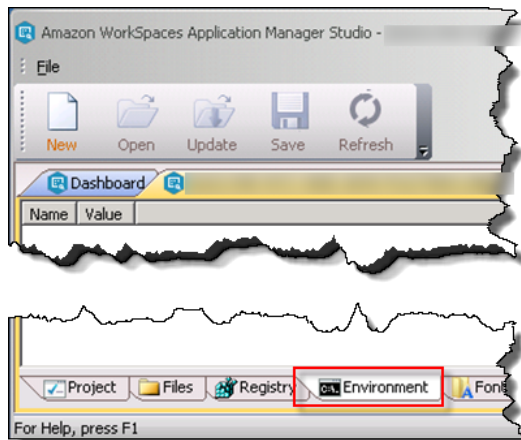
Registry keys that are excluded in a package are not used in Amazon WorkSpaces, but are still part of the package. This gives you the flexibility of including specific keys one package deployment and not for another package. Excluding keys also enables you to test a package without a key before removing the key from the project.

1. In the **Registry** tab, open the context (right-click) menu for the root or subkey that you want to exclude, and then choose **Exclude**.
2. Choose **OK**.

Excluded keys appear in italicized text.

## Working with Environment Variables

You can set the environment variables used by your package by choosing the **Environment** tab in the project dashboard. You can see the environment variables in the package, add new environment variables, or change existing environment variables.



The **Environment** tab shows two types of environment variables: user environment variables (specific for each user and set in the registry under `HKEY_CURRENT_USER \ Environment`) and system environment variables (global for everyone and set in the registry under `HKEY_LOCAL_MACHINE \SYSTEM\CurrentControlSet\Control\Session Manager\Environment`). These environment variables are set in the application or on the operating system depending on the **Environment Variables** setting in **Virtualization** in the **Settings** dialog box.



- Environment variables shared on the entire operating system are added to the `HKEY_CURRENT_USER\Environment` registry key when you create the package. These environment variables are available to all applications for that operating system.
- Environment variables visible only to the application are set in the process environment when the application starts in the operating system. Child processes of the application may inherit these environment variables but other process on the system cannot access these environment variables.

All virtual environment variables override any existing variables of the same name for either method. The only exception to this rule is for the `%PATH%` or `%PATHEXT%` variables. These variable are automatically appended to existing system values. For example:

```
PATH = %PATH%;c:\newfolder
```

For more information about setting environment variables, see [How to propagate environment variables to the system](#) in the Microsoft Knowledge Base.

#### **Note**

Some legacy applications may check the registry for environment variables under a specific hive and not use the variables in the process environment. For such applications, add the environment variables directly using the **Registry** tab in the correct locations and set the key disposition to Virtual-isolated (layer 4).

#### **To add an environment variable**

1. In the **Environment** tab, open the context (right-click) menu in the pane, and choose **Add Environment Variable**.
2. In the **Add Environment Variable** dialog box, enter the name and value of the environment variable, and choose **OK**.

#### **To edit an environment variable**

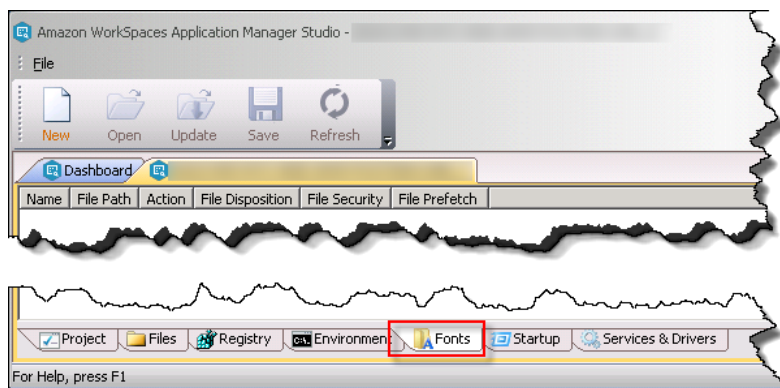
1. In the **Environment** tab, open the context (right-click) menu for the environment variable that you want to edit, and choose **Edit**.
2. In the **Edit Environment Variable** dialog box, edit the name and value of the environment variable, and choose **OK**.

#### **To delete an environment variable**

- In the **Environment** tab, open the context (right-click) menu for the environment variable that you want to delete, choose **Delete**, and choose **OK**.

## Working with Fonts

You can configure the fonts used in your package by choosing the **Fonts** tab in the project dashboard.



The fonts in a package are either available only to the applications in the package or made available to other applications on the operating system, depending on the **Font** setting under the **Virtualization** in the **Settings** dialog box.

#### Note

To add more fonts to a project, you need to know the file name or names for the font, and not just the font name. In many cases, there is more than one file for each font. For example, Arial consists of four files, one for the basic font, and others for italic, bold, and bold italic.

#### To import a system font

1. In the **Fonts** tab, open the context (right-click) menu in the pane and choose **Import Font**.
2. In the **Import Font** dialog box, select a font, and choose **OK**.

Repeat this step to import other system fonts.

#### To add a font from a font file

1. Copy the font file to the package instance.
2. In the **Fonts** tab, open the context (right-click) menu in the pane, and choose **Import Font**.
3. In the **Add Font** dialog box, enter a name for the font, and choose **Browse** to find the font file.
4. Choose **OK**

#### Note

If the virtualization setting for fonts is set to register, but the disposition for the corresponding font file is set to Virtual-isolated (layer 4), the disposition for the font appears in red. This font is not visible to other applications on the operating system, despite the virtualization setting.

#### To edit a font

1. In the **Fonts** tab, open the context (right-click) menu for the font that you want to edit, and choose **Edit**.
2. In the **Edit Font** dialog box, change the values, and choose **OK**.

#### To change the properties of a font

You can edit the prefetch, disposition, and security settings from **Fonts** tab, instead of finding and editing the font file in **Files** tab.

1. In the **Fonts** tab, open the context (right-click) menu for the font that you want to edit and choose **Font File Properties**.

2. In the **Font File Properties** dialog box, change the property values, and choose **OK**.

### To delete a font

- In the **Fonts** tab, open the context (right-click) menu for the font that you want to delete, choose **Delete**, and choose **OK**.

### To exclude a font

Fonts that are excluded are not used in Amazon WorkSpaces, but are still part of the project. This gives you the flexibility of including specific fonts in one package and not in another package. Excluding fonts variables also enables you to test a package without font before removing the font files from the package.

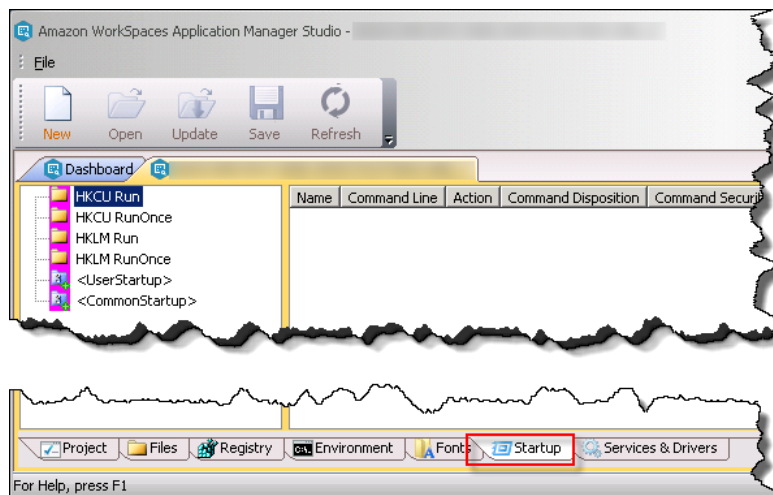
1. In the **Fonts** tab, open the context (right-click) menu for the font that you want to exclude, and then choose **Exclude**.
2. Choose **OK**.

Excluded fonts appear in italicized text.

## Working with Startup Items

Startup items are programs that typically launch at the operating system startup, which are installed along with the application. These often include the system tray icons that permit direct access to the functions of the application, even when the application is not running.

You can configure the fonts used in your package by choosing the **Startup** tab in the project dashboard.



The operating system launches startup items as either as a special registry key called run keys or as a Start item. You can set your launch preference in **Virtualization** of the **Settings** dialog box.

The path to the executable in a run key is displayed in the **Startup** tab.

**HKCU Run** (HKCU is short for HKEY\_CURRENT\_USER)  
Applies to the current user.

### HKCU RunOnce

Applies to the current user but only runs once.

### HKLM Run (HKLM is short for HKEY\_LOCAL\_MACHINE)

Applies to all Amazon WorkSpaces.

To add a startup item, you need following information:

- If you want the startup item to launch as a registry key or as a Start menu item
- Name and path of the registry key (if necessary)
- Path and name of the executable file

### To add a registry key startup item

1. In the **Startup** tab, open the context (right-click) menu for either the **HKCU** or **HKLM** folder in the left navigation pane and choose **Add Startup Value**.
2. In the **Add Startup Value** dialog box, enter the name and path to the executable file of the startup item, and choose **OK**.

### To add a Start menu startup item

1. In the **Startup** tab, open the context (right-click) menu for either the **<UserStartup>** or **<CommonStartup>** folder in the left navigation pane and choose **Add Startup File**.
2. In the **Add Startup File** dialog box, enter the name and path to the executable file of the startup item. To use a different name for the executable file, choose **Override source file name with the following name** and enter the new name.
3. Choose **OK**.

### To edit a startup item

1. In the **Startup** tab, open the context (right-click) menu for the font that you want to edit, and choose **Edit**.
2. In the dialog box, change the values and choose **OK**.

#### Note

If the disposition of the registry key for a startup item is Installed-permanent (layer 1) or Installed-temporary (layer 2), but the corresponding executable file disposition is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4), the item is flagged with the disposition in red, because the registry key is physically present on the client, while the file to which it points is not.

### To change the properties of a startup item

You can edit the prefetch, disposition, and security settings from **Startup** tab, instead of finding and editing the font file in **Files** tab.

1. In the **Startup** tab, open the context (right-click) menu for the font that you want to edit, and choose **Edit Command Line File Properties**.
2. In the **Command Line File Properties** dialog box, change the property values, and choose **OK**.

#### Note

If a startup item points to a shortcut (.lnk file) which has a Installed-permanent (layer 1) or Installed-temporary (layer 2) disposition, but the target executable file is set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4), the item is flagged with the disposition in red,

because the shortcut is physically present in the operating system, while the file to which it points is not.

### To delete a startup item

- In the **Startup** tab, open the context (right-click) menu for the font that you want to delete, choose **Delete**, and choose **OK**.

### To exclude a startup item

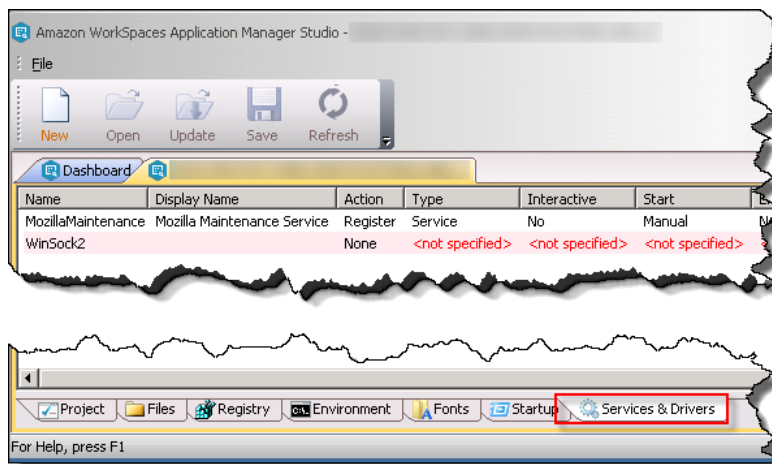
Startup items that are excluded are not used in Amazon WorkSpaces, but are still part of the package. This gives you the flexibility of including specific startup items in one package deployment and not in another package. Excluding startup items also enables you to test a package without a startup item before removing the startup from the package.

1. In the **Startup** tab, open the context (right-click) menu for the font that you want to exclude, and then choose **Exclude**.
2. Choose **OK**.

Excluded fonts appear in italicized text.

## Working with Services and Drivers

You can view the services and drivers installed when the package was created or add new services and drivers by choosing the **Services & Drivers** tab in the project dashboard.



Driver and services are shared components within the operating system, which means that while the files and registry keys can be virtual, the running process affect the entire operating system. Certain types of services or drivers must be ran at system startup and these, as well as any dependencies, may need to be made to Installed-temporary (layer 2). When you remove a service or driver, you may have to use a configurable AppEvent trigger to perform a system reboot.

### To add a service or driver

1. In the **Services & Drivers** tab, open the context (right-click) menu in the pane, and choose **Import Service or Driver**.
2. In the **Import Service or Driver** dialog box, select a service or driver from the list and choose **OK**.
3. Choose **Yes** to add the image for the service or driver to your package.

### To change the disposition of the registry key for a service or driver

1. In the **Services & Drivers** tab, open the context (right-click) menu of the service or driver for which you want to change the disposition, and choose **Service Key Disposition**.
2. In the **Service Key Disposition** dialog box, select the disposition for the registry key, and choose **OK**. If the message box appears, choose **Yes** to change the ancestor keys to Virtual-integrated (layer 3).

### To edit the image file properties for a service or driver

If the you added the image file for service or driver through the **Files** tab, you can edit the prefetch, disposition, and security settings from within the **Services** tab, rather than having to locate and edit the file from the **Files** tab.

#### Note

Any service or driver in a package with Virtual-integrated (layer 3) disposition are guaranteed to have all of their dependencies (e.g. .dll, .ini, .manifest, .cfg, etc. files) available to the service or driver. If you set the service or driver to Installed-permanent (layer 1) or Installed-temporary (layer 2), set the dependencies to the Installed-permanent (layer 1) or Installed-temporary (layer 2) disposition. This is important if the services are meant to automatically start at Windows logon. You can use Dependency Walker as a tool to identify DLL files that an executable needs to run successfully.

1. In the **Services & Drivers** tab, open the context (right-click) menu for the service or driver for which you want to change the image file properties and choose **Edit Image File Properties**.
2. In the **Image File Properties** dialog box, select your settings, and choose **OK**.

### To edit the service or driver properties

1. In the **Services & Drivers** tab, open the context (right-click) menu for the service or driver for which you want to change the image file properties, and choose **Edit**.
2. In the dialog box, change the settings and choose **OK**.

### To change the failure action of a service or driver

- In the **Services & Drivers** tab, open the context (right-click) menu for the service or driver for which you want to change the failure action, and choose **Turn Failure Actions OFF** if restart is enabled or **Turn Failure Actions ON** if restart is already disabled. The option is grayed out if the failure action cannot be changed for the selected service or driver. If the failure actions are enabled for a service or driver, the name of that service is red.

### To delete a service or driver

- In the **Services & Drivers** tab, open the context (right-click) menu for the service or driver that you want to delete, choose **Delete**, and choose **OK**.

### To exclude a service or driver

Services or drivers that are excluded are not used in operating system, but are still part of the package. This gives you the flexibility of including specific services or drivers in one package deployment and not in another package. Excluding services or drivers also enables you to test an package without a service or driver before removing the service or driver from the package.

1. In the **Services & Drivers** tab, open the context (right-click) menu for the service or driver that you want to exclude, and then choose **Exclude**.

2. Choose **OK**.

Excluded services or drivers appear in italicized text.

## Adding Required Applications or Scripts

Some applications may require additional applications or scripts to run in order for the packaged application to run properly. These additional applications or scripts may need to be launched before, during, or after the packaged application launches. For example, you may need these additional applications or scripts to do the following:

- Configure plug-ins for office suites and browser applications
- Configure virtual printers
- Start a background application before the packaged application

To launch these additional applications or scripts, you can use a configurable AppEvent trigger. A configurable AppEvent trigger is an action you set in Amazon WorkSpaces Application Manager Studio that runs a handler to launch these additional applications or scripts. The handler can be an application written in any programming language, such as C++, or a script in a scripting language, such as VBScript.

Applications used in a configurable AppEvent trigger can be any size while scripts have to be less than 10 Mb in size. Applications or scripts added to the package cannot be directly modified. If the application or script is modified in the package, Amazon Studio or Amazon Admin Player disables the application or script.

## Sample Applications and Scripts

The packaging instance contains the following sample applications and scripts in `C:\Program Files\Amazon\WAM Admin Studio\Samples`.

Filename	Description	Error codes
AddService.exe	Adds and starts a service	-1
DotNetCheck.vbs	Checks for the Microsoft .NET Framework. Specify the major and minor version number to check in <b>Parameter</b> .	1
DirectXCheck.vbs	Checks for a specific version of Microsoft DirectX. Specify the major and minor version number to check in <b>Parameter</b> .	1
OpenFirewall.vbs	Opens the firewall for the specified program or ports.	-1
OSMemCheck.vbs	Check for sufficient memory.	-1
QuicktimeCheck.vbs	Checks for specific version of Apple QuickTime. Specify the major and minor version number to check in <b>Parameter</b> .	1
ReaderCheck.vbs	Checks for a specific version of Adobe Reader. Specify	1

Filename	Description	Error codes
	the major and minor version number to check in <b>Parameter</b> .	
RemoveService.exe	Stops and removes a service	-1

## Handler Errors

If the handler returns a failure code, Amazon Studio logs the failure code but does not display any messages for error codes. The handler must display a message for error codes. If the failure occurs during the activation trigger, the application is not added to the Amazon Admin Player. If the failure occurs during the exit trigger, the handler continues.

### Note

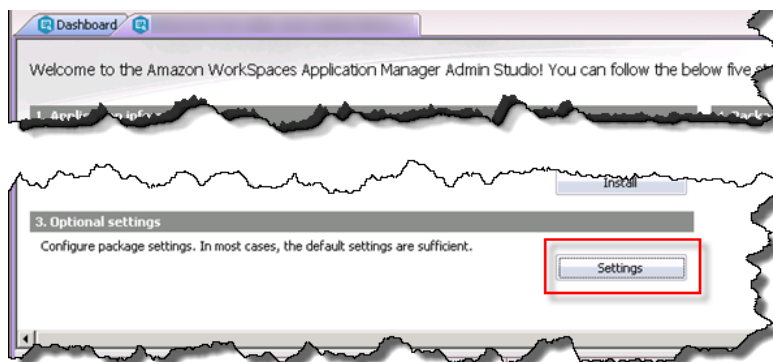
During a Windows logoff or shutdown, triggers are skipped because Windows does not allow a new process to be created during this time.

## Adding a Configurable AppEvent Trigger

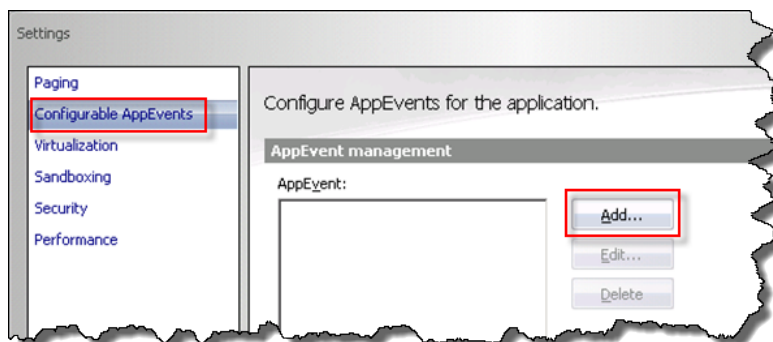
You can add a configurable AppEvent trigger through Amazon Studio.

### To add a configurable AppEvent

1. In Amazon Studio, choose **Update** to get the files in your package.
2. Under **3. Optional settings**, choose **Settings**.



3. In the navigation pane, choose **Configurable AppEvents** and then choose **Add**.



4. In the **Configurable AppEvent Settings** dialog box, do the following:



- For **Trigger**, select when to launch the handler.

Trigger	Description
<b>After registration</b>	Occurs one time when the application is first added to the Amazon WorkSpaces Application Manager client application.
<b>After activation</b>	Occurs one time immediately after the application package has been activated (installation is triggered).
<b>After virtualization</b>	Occurs immediately after virtualization has completed (installation is complete).
<b>After launch</b>	Occurs immediately after the application launches.
<b>After exit</b>	Occurs after the application exits.
<b>Before deactivation</b>	Occurs one time immediately before an application package deactivates and the application is removed from the Amazon WorkSpaces Application Manager client application.
<b>Before devirtualization</b>	Occurs before devirtualization starts and the application is removed from the Amazon WorkSpaces.

- For **Frequency**, choose when to launch the trigger.

Trigger	Description
<b>Fire Once</b>	The action only launches one time.
<b>Fire Always</b>	The action launches when the trigger occurs. If the trigger occurs only one time, then this action launches only one time.

- For **Handler command line**, enter the path and required parameters for the handler. If the application or script is on Amazon WorkSpaces, choose **Use local application**.

**Caution**

When using the **Extra** parameter, be sure to not include the <, >, |, &, ^, ", #, %, {, }, \, ~, [, ], ', or space characters unless they are encoded by the portal and decoded by the handler. All characters appear as part of the value in the extra environment variable and do not need to be escaped.

- For **Handler launching**, choose one of the following as required:

**Note**

This setting can affect critical files and should be used cautiously.

Run elevated (only supported on Windows Vista or later)

If the handler requires administrator privileges.

Run as LocalSystem (only if allowed by group policy)

If the handler should be launched as a system process. The application or script in the handler has same access to files as the user.

- For **After launching**, choose **Wait for handler** to exit to require the handler to finish before the configurable AppEvent trigger in the Amazon WorkSpaces Application Manager client application goes to the next state. If this option is cleared, then the handler launches and the trigger continues with the normal operation in parallel. Set the **Time out after** in minutes or seconds for the trigger in the client to wait for the handler to exit. In **Success codes**, set a specific return value to indicate success or failure for the trigger. By default, 0 is a success code and all other values are failures. You can also set a different success code.
- Choose **OK** to add the configurable AppEvent trigger.

## Sandboxing Your Package

Sandboxing means to separate changes made to the file system or registry from the Amazon WorkSpaces Application Manager client application. Amazon WAM Admin Player performs partial sandboxing for files, which means that only content modified or added to folders or subfolders of root folders in the package are sandboxed. For example, if a package contains only the installation root folder and the **Program Files** folder, and if the application creates a new file under the **Windows** folder, then this new file is directly written to the system. Amazon Admin Player sandboxes all default registry hives (that is, HKCR, HKCU, HKLM, HKU), but any other hive must be part of the package to be sandboxed.

For the paths in the package that are sandboxed, the content is separated from the system, meaning that they are not written to the local system but stored separately and only made available to the application. Again using the example above, if the package creates a new file under the **Program Files** folder, then the file is sandboxed and not written to the system. Only existing files under the **Program Files** folder that are not contained in the package would be modified by the application if attempted. The same rule applies for existing registry keys not contained within the package.

As a rule, do not sandbox any documents or data saved by the user. The following table lists the folders that are the only exception for sandboxed folders.

### Sandbox exclusion folders

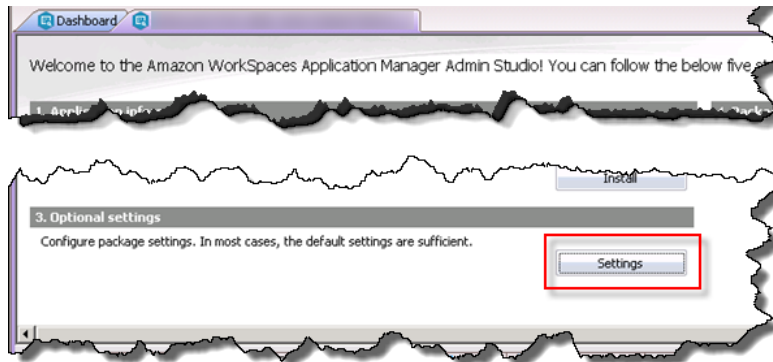
Sandbox exclusion folder	Path
System Root	C:\
Desktop	C:\Users\ <i>&lt;username&gt;</i> \Desktop
Documents	C:\Users\ <i>&lt;username&gt;</i> \Documents
Downloads	C:\Users\ <i>&lt;username&gt;</i> \Downloads
Music	C:\Users\ <i>&lt;username&gt;</i> \Music
Pictures	C:\Users\ <i>&lt;username&gt;</i> \Pictures
Videos	C:\Users\ <i>&lt;username&gt;</i> \Videos
Links	C:\Users\ <i>&lt;username&gt;</i> \Links
Favorites	C:\Users\ <i>&lt;username&gt;</i> \Favorites
Contacts	C:\Users\ <i>&lt;username&gt;</i> \Contacts
Saved Games	C:\Users\ <i>&lt;username&gt;</i> \Saved Games
Searches	C:\Users\ <i>&lt;username&gt;</i> \Searches

Sandbox exclusion folder	Path
Temp	C:\Users\ <i>&lt;username&gt;</i> \AppData\Local\temp
Internet Cache	C:\Users\ <i>&lt;username&gt;</i> \AppData\Local\Microsoft\Windows\Temporary Internet Files
Shared Documents	C:\Users\ <i>&lt;username&gt;</i> \Documents
Shared Downloads	C:\Users\ <i>&lt;username&gt;</i> \Downloads
Shared Music	C:\Users\Public\Music
Shared Pictures	C:\Users\Public\Pictures
Shared Video	C:\Users\Public\Videos

If your application writes to these folders, the data is directly written to the system, even if the **Users** folder is part of the application as a root folder, as would be expected by users.

### To configure sandboxing in your package

1. In Amazon Studio, choose **Update** to get the files in your package.
2. Under **3. Optional settings**, choose **Settings**.



3. In the navigation pane, choose **Sandboxing**.
4. To exclude a folder, choose **Folder Exclusions**. In the **Sandbox Folder Exclusions** dialog box, choose **Add** and then type the path to the folder that you want to exclude. Choose **OK**.

#### Tip

You can drag-and-drop the folder to exclude to the **Folder Exclusions** dialog box from Explorer.

5. To exclude a registry key, choose **Registry Key Exclusions**. In the **Sandbox Registry Key Exclusions** dialog box, choose **Add** and enter the registry key to exclude from the package.
6. For **Disposition for new registry key roots**, choose **Virtual-integrated (layer 3)** for resources that the application and local system can see but are not physically installed on the local system, or **Virtual-isolated (layer 4)** for resources that only the application can see and are not physically installed on the local system.
7. For **Application settings**, choose **Preserve application settings when application is removed** to save the application-specific settings modified by the user when the application assignment is removed from the user, or **Allow Windows to read application settings** to write the application-specific settings to the standard Windows roaming profiles location.

8. Choose **OK**.

## Setting File Security

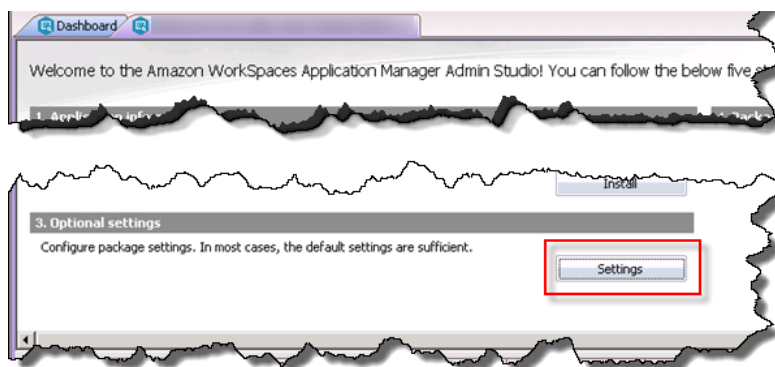
The standard security settings for the files in your application may need to be bypassed by specific application processes. For example, an application patch (such as Windows Updates) may be required to modify a file with a read-only attribute. In this case, a security override can be used to allow this to happen while still protecting the application. Another example would be to not allow anti-virus or indexing software from fetching all the application pages. This undesirable side effect can also be prevented by denying those specific processes from accessing the virtual application.

### Note

Using a different security setting only affect the file security settings but do not change the visibility of the disposition layer. If the file disposition is set to **Virtual-isolated (layer 4)**, then setting a system process to override all the security settings still does not allow that process to see this file.

### To change the security of a file

1. In Amazon Studio, choose **Update** to get the files in your package.
2. Under **3. Optional settings**, choose **Settings**.



3. In the navigation pane, choose **Security** and **Add**.
4. In the **Security Override Process** dialog box, do the following:
  - For **Process name**, type the exact name of the process for the security override controls. Do not use wildcards.
  - For **Recognition method**, choose one of the following methods to use to determine which file version to override the process. You may have several versions of the same file in the system. For any option that requires a file path, enter the path and file name in the **Process file** field.
    - Process name  
Use only the process name. This is the weakest verification method.
    - Process path  
Use the path and filename of the process.
    - MD5 Hash  
Use an MD5 algorithm to compute and store a fingerprint of the executable file to verify that the process file is authentic. This method is the most secure verification method.
    - CRC Checksum  
Use a CRC algorithm to compute and store a checksum value to verify that the process file is authentic.
  - For **Choose how virtual assets should be accessed (disposition override)**, choose one of the following:

Allow or deny access based on the dispositions specified for the virtual assets  
Use the individual disposition setting for files, folders, registry keys, and values.

Always allow access

Treat all files, folders, registry keys, and values as having Virtual-integrated (layer 3) disposition, even if they have Virtual-isolated (layer 4) disposition.

Always deny access

Treat all files, folders, registry keys, and values as having Virtual-isolated (layer 4) disposition, even if they have Virtual-integrated (layer 3) disposition.

- For **Security settings override**, choose **Allow** or **Deny** for the following options and choose **Apply**.

**Note**

**Deny** takes precedence over **Allow**. Choosing both settings is the same as choosing **Deny**.

Read or copy content of files

Read or copy the application files.

Write or modify content of files

Write or modify the contents of the application files.

Make files visible through folder listings

Show the files in the folder listing.

## Finding a Package with Amazon WAM

You can see a list of packages in the application catalog through the Amazon WorkSpaces console.

### To find a package

1. Sign in to the AWS Management Console and open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/>.
2. Choose **Packages**, which shows a list of all of the packages in the application catalog.

## Sharing a Package with Amazon WAM

You can create an application package and then share it with other AWS accounts. Accounts with access to your shared package can create a new application package that is based on your shared application.

Your packages are shared by version. To share a new version of a package, add the AWS account number to the permission list of the new package.

### To share a package

You need the AWS account numbers for the accounts with which you want to share the package.

1. Sign in to the AWS Management Console and open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/>.
2. Choose **Packages**, which shows a list of all of the packages in the application catalog.
3. Select your package and choose **Modify Permissions**.
4. For **Version**, select the version of the package to share.
5. For **AWS Account Number**, type the AWS account number of the account with which to share the package and choose **Add Permission**.

6. Choose **Save**.

After you add it to the permission list, the other account can upload the application using the shared package to their application catalog by choosing **Applications** under **Application Manager** in their console, choosing **Upload new application** and then following the steps. For more information, see [Uploading Your Own Applications \(p. 10\)](#).

#### To remove an account from the permissions list

1. Sign in to the AWS Management Console and open the Amazon WorkSpaces console at <https://console.aws.amazon.com/workspaces/>.
2. Choose **Packages**, which shows a list of all of the packages in the application catalog.
3. Select your package and choose **Modify Permissions**.
4. Next to the account number to delete, choose **Delete** and **Save**.

## Deleting a Package with Amazon WAM

You can delete a package through the Amazon WorkSpaces Application Manager Studio.

#### Note

When you delete a package in Amazon WorkSpaces Application Manager Studio, you delete all versions of the packages.

You cannot delete a package if it has been shared with other AWS accounts within the region or has apps created and provisioned from it. To delete this kind of package, unshare the package and delete all apps associated with it.

#### To delete a package

1. Connect to your packaging instance using a remote desktop client and launch Amazon WorkSpaces Application Manager Studio. The remaining instructions are performed on the package instance.
2. Select a package from the **Dashboard**, and then choose from one of the following options:
  - Open the context menu (right-click) and choose **Delete**.
  - Choose **Delete** in the toolbar.
  - Open the package to view the package details, and then choose **Delete** in the toolbar.

## Troubleshooting Amazon WAM Packaging Issues

The following are possible issues that you might have when packaging an application.

#### Issues

- [Finding the Log Files \(p. 58\)](#)
- [Application Packaging Issues \(p. 58\)](#)
- [Device Driver and Service Issues \(p. 61\)](#)
- [Missing Shortcut, File, and Folder Issues \(p. 63\)](#)
- [Other Issues \(p. 65\)](#)
- [User Account Issues \(p. 66\)](#)

## Finding the Log Files

If you are running into packaging issues, check the Amazon WorkSpaces Application Manager Studio and the Amazon WAM Admin Player log files.

### To see the Amazon WorkSpaces Application Manager Studio log file

The Amazon Studio log file contains every error that occurred when you packaged your application.

- In the `C:\Program Files\Amazon\WAM Admin Studio` folder of the packaging instance, open the `WamAdminStudio.log` file.

### To see the Amazon Admin Player log file

The Amazon Admin Player log file shows the files and registry keys captured and filtered to the package and any errors.

1. In the Amazon WAM Admin Player, choose **File** and **Options**.
2. In the **Options** dialog box, on the **Log** tab, choose **View Log**.

## Application Packaging Issues

The following are possible application packaging issues.

### Issues

- [How can I hide the Add/Remove Programs entry for my application? \(p. 58\)](#)
- [I cannot change the settings in my package \(p. 59\)](#)
- [I cannot print from my application \(p. 59\)](#)
- [I cannot retrieve my application list \(p. 59\)](#)
- [My application does not run because of a licensing issue \(p. 59\)](#)
- [My application fails to start \(p. 59\)](#)
- [My application is missing dependencies \(p. 60\)](#)
- [My file associations do not work \(p. 60\)](#)

## How can I hide the **Add/Remove Programs** entry for my application?

Many applications add an entry in the **Add or Remove Programs** or **Programs and Features** list in the Windows Control Panel. This entry can be confusing because the application is virtual and cannot be uninstalled. You can remove this entry from the list.

### To hide the application entry

1. In the project in Amazon WorkSpaces Application Manager Studio, choose the **Registry** tab.
2. Open the context (right-click) menu of the following registry key and choose **Delete**.

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall
```

## I cannot change the settings in my package

You opened your package in read-only mode by double-clicking the package in the dashboard. To change the settings in your the package, open the package in update mode by choosing **Update** on the Amazon Studio toolbar. If the application in the package is already installed on the packaging instance, a message box offers a choice to overwrite the existing files. Choose **Yes** to overwrite the files on the package instance with the files in the package. The overwritten files are displayed in the **Progress** dialog box.

## I cannot print from my application

Some application install new printer devices, which can be included in a package. On the **Files** tab of Amazon WorkSpaces Application Manager Studio, the **spool** folder is a subdirectory found under the **SystemDir** system folder. Applications that come with printer drivers may not work properly if this folder is removed from the **Files** tab. On the **Registry** tab, the following registry key contains references to the printer drivers in the `spool` folder:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Print\Environments  
\Windows NT x86\Drivers
```

If a new printer does not appear in the application, use the following procedure.

### To restart the print spooler

1. In Amazon WorkSpaces Application Manager Studio, choose the **Project** tab on the top of the project pane.
2. Under **3. Optional settings**, choose **Settings**.
3. In the **Settings** dialog box, choose **Virtualization**.
4. Under **Special requirements**, choose **Restart Print Spooler** and **OK**.

## I cannot retrieve my application list

If you receive an error from the Amazon WorkSpaces Application Manager Studio stating that your app list could not be retrieved, verify that you have created the proper IAM role as explained in [Create the Application Packaging Role \(p. 3\)](#), and that you applied that role to the packaging instance.

## My application does not run because of a licensing issue

Many applications employ some form of copy protection to prevent theft and enforce proper licensing. Amazon WorkSpaces Application Manager Studio does not remove copy protection. If you need to enable multiple users to use an application, package the application so that it uses a serial number for high volume licensing and multiple systems can activate with a single key (for example, Microsoft products use volume license keys). Otherwise, the application prompts users for a license; not all systems will be able to activate and make the application fully functional.

Test the package on other systems to ensure that there is no form of license enforcement that you are not already aware of. If the package does not work or prompts for licensing, retest the package on the packaging instance. If the application binds to hardware, it should work on the same system. For information about supported licensing models, contact the software ISV.

## My application fails to start

Sometimes an application can fail to start when the command line is incorrect. You might have to specify a different executable at the command line or you might need to add parameters. Check the original application shortcut, typically found on the **Start** menu, and use the same command line and



parameters as what appears in **4. Package build** of the Amazon WorkSpaces Application Manager Studio.

Another problem may be that some applications modify the `PATH` environment variable of the local machine during installation or after starting for the first time. There may be some system-specific entries included in the `PATH` variable (for example: `PATH=%PATH%;C:\ApplicationPath1;C:\ApplicationPath2`). Ensure that the package does not contain system-specific entries by choosing the **Environment** tab, **Path**, and then checking for the any application-specific path environment variables in the **Value** field.

## My application is missing dependencies

Many applications require the presence of a specific runtime library or a minimum version of the .NET Framework. Amazon WorkSpaces Application Manager Studio comes with the latest builds of the following C++ runtime libraries:

- Visual C++ 2008 x86 and x64
- Visual C++ 2005 x86 and x64

If your application requires other dependencies, install the dependencies before you install the application. After you capture the dependency installation, set the disposition of the dependency files and registry keys to **Virtual-isolated (layer 4)** before you upload your package.

### To set the disposition of files and registry keys

1. In Amazon WorkSpaces Application Manager Studio, choose the **Files** tab on the top of the project pane.
2. Open the context menu (right-click) for the dependency file or folder and choose **Properties**.
3. In the **Properties** dialog box, choose **Virtual-isolated (layer 4)** and **OK**.
4. On the **Registry** tab, open the context menu (right-click) for the registry hive or key and choose **Disposition**.
5. In the **Registry Key Disposition** dialog box, choose **Virtual-isolated (layer 4)** and **OK**.

## My file associations do not work

Windows manages file extension associations to applications under the `HKEY_CLASSES_ROOT` hive of the Windows registry. The file type must be set in the registry and the application executable must not be hidden. Make sure that the file extensions you need exist in the **Registry** tab of the project. If the extension is missing, manually add the extensions.

You can also package a new version of the application where you choose **All processes** under **Capture file and registry activity for the following processes** in the **Capture Application Installation** dialog box, run the application, and configure the default file associations. Check that the application executable file is not hidden and the disposition is set to **Virtual-integrated (layer 3)**.

### To create a new package version

To create a new package version, you need the installation files.

1. In Amazon WorkSpaces Application Manager Studio, choose the **Dashboard** tab on the top of the project pane, select your project in the list, and then choose **Update** in the toolbar and **Close**.
2. Under **2. Application installation**, choose **Install**.
3. In the **Capture Application Installation** dialog box, do the following:
  - For the **Installer path** field, enter the path and file name to your installer.

- For the **Command line parameter** field, enter any parameters required to install your application.
  - Under **Capture file and registry activity for the following processes**, choose **All processes**.
  - Choose **Ignore changes under the Installer path specified above**.
  - Choose **Launch**, install your application, and then choose **OK** and **Close**.
4. On the **Files** tab, open the context (right-click) menu for your executable file and choose **Properties**.
  5. In **File Properties**, under **Disposition**, choose **Virtual-integrated (layer 3)** and **OK**.
  6. On the **Project** tab, do the following:
    - In **4. Package build**, enter text to describe this new version in the **Label** field. This text appears as a version of the package.
    - In **5. Package upload**, choose **Upload**.

## Device Driver and Service Issues

The following are possible device driver and service issues.

### Issues

- [How do I package drivers for Plug and Play devices? \(p. 61\)](#)
- [I cannot stop a service with a running status \(p. 62\)](#)
- [My Windows services or drivers do not start \(p. 62\)](#)

## How do I package drivers for Plug and Play devices?

Plug and play drivers for devices such as printers are installed in two phases:

- Amazon WorkSpaces Application Manager Studio uses the installer to copy the driver files to the hard drive.
- Windows associates the files with the correct device.

Starting with Vista, Windows uses a protected repository for trusted device drivers called the driver store. To set up a device drive, use a Configurable AppEvent that uses the `pnputil.exe` utility. This utility adds and removes drivers from the driver store. The syntax for adding drivers using the utility is:

```
pnputil.exe -i -a "C:\<folder>\driver.inf"
```

### Caution

Do not attempt to set files in the driver store to Installed-permanent (layer 1) or Installed-temporary (layer 2) because it causes virtualization to fail.

### To exclude device driver files

1. In Amazon WorkSpaces Application Manager Studio, choose the **Files** tab on the top of the project pane.
2. Open the context menu (right-click) for the `<WindowsDir>\inf\` folder and choose **Exclude**.
3. To exclude all virtual files used by the device driver, do the following:
  - On the instance desktop, do the following to open Device Manager:
    - Choose **Start**.
    - type `devmgmt.msc` and then press Enter.
  - Double-click the device driver in the list to open the **Properties** dialog box.

- In **Properties**, choose the **Driver** tab and **Driver Details**, make note of the files listed in the **Driver File Details** dialog box, and then choose **OK**.
  - In Amazon WorkSpaces Application Manager Studio, choose the **Files** tab on the top of the project pane.
  - Open the context menu (right-click) for each file and choose **Exclude**.
4. To exclude files for TWAIN devices, do the following:

#### To exclude files for TWAIN devices

1. In the **Dashboard** tab of Amazon WorkSpaces Application Manager Studio, select your project and choose **Update** on the toolbar.
2. Choose the **Files** tab.
3. For each driver file under `<WindowsDir>\SSDriver`, open the context menu (right-click) and choose **Properties**.
4. In the **Properties** dialog box, choose **Installed-temporary (layer 2)** under **Disposition** and **OK**.
5. In Amazon WorkSpaces Application Manager Studio, select your project on the top of the project pane, and then under **3. Optional settings**, choose **Settings**.
6. In **Settings**, choose **Configurable AppEvents** and **Add**.
7. In the **Configurable AppEvent Settings** dialog box, choose **After activation** under **Trigger**.
8. • In **File name** in **Handler command line** type `%SystemRoot%\system32\pnputil.exe`. Under **Handler Launching**, choose **Run elevated (only supported on Windows Vista or later)** and **OK**.

#### Note

Device drivers remain on the system even after the application has been removed by Amazon WAM Admin Player. This is expected behavior.

## I cannot stop a service with a running status

Windows Task Manager tracks all executable files and services with Virtual-integrated (layer 3) or Virtual-isolated (layer 4) disposition. When the executable file or service is in use, Windows Task Manager shows this file with a **Status** value of **Running**.

To avoid usage tracking, set the disposition of these files (such as Service ImagePath) and services to Installed-temporary (layer 2) disposition.

In addition, some services may be configured to restart automatically after an interruption.

#### To configure an interrupted service to restart automatically

1. In Amazon WorkSpaces Application Manager Studio, choose the **Registry** tab on the top of the project pane.
2. Remove the Failure action key from the following hive for the service:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
```

## My Windows services or drivers do not start

If a service needs to start using a unique user account (this normally defaults to LocalSystem), the application needs a Configurable AppEvent to create this account.

### Caution

Some systems services have security descriptors that are strictly enforced. When adding a new service, it is important to match the security descriptor of the installed version. Use `sc sdshow <service_name>` at the command prompt to display the security descriptor set for the service. Marking an application to require a system restart after adding a service typically corrects the security descriptors.

If the application has boot or system drivers, mark all of the files and registry keys for those services or drivers as Installed-temporary (layer 2).

### To set the disposition of the files and registry keys

1. In Amazon WorkSpaces Application Manager Studio, choose the **Services & Drivers** tab.
2. Open the context menu (right-click) for the service or driver and choose **Service Key Disposition**.
3. In the dialog box, choose **Installed-temporary (layer 2)**, and **OK**.
4. Open the context menu (right-click) for the service or driver and choose **Image File Properties**.
5. In the dialog box, choose **Installed-temporary (layer 2)**, and **OK**.

### Caution

Set both the files and registry keys to Installed-temporary (layer 2) disposition. You may have problems if the files and registry keys do not have the same disposition setting.

6. On the **Registry** tab, open the context menu (right-click) for each legacy registry key for your service or driver and choose **Exclude**.
7. Open the context menu (right-click) for each Enum registry key for your service or driver and choose **Exclude**.
  - HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Enum\Root
  - HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Services\{NAME}\Enum
8. On the **Project** tab, choose to allow or avoid a system restart. In **3. Optional settings**, choose **Settings**, and then do one of the following in the **Settings** dialog box:

### To require a system restart

1. Choose **Virtualization**.
2. Under **Special requirements**, select **Require system reboot** and **OK**.

### To avoid a system restart

Use a Configurable AppEvent to avoid a system restart.

1. Choose **Configurable AppEvents** and **Add**.
2. In the **Configurable AppEvent Settings** dialog box, choose **After Launch** under **Trigger**.
3. In **File name** under **Handler command line**, type `sc.exe start <driver_name>` and **OK**.

For more information about how services work, go to [How to create a Windows service](#) in the Microsoft Knowledge Base.

### Note

A system driver cannot be stopped and removed from a system without a reboot. Amazon WorkSpaces Application Manager Studio removes these files from the system but the driver is still loaded until a reboot occurs.

## Missing Shortcut, File, and Folder Issues

The following are possible issues with missing shortcuts, files, and folders.

## Issues

- [My shortcut is missing \(p. 64\)](#)
- [My shortcut is a blank window \(p. 64\)](#)
- [The text is in the wrong font \(p. 64\)](#)
- [My folders and files are not in the application in my WorkSpace \(p. 65\)](#)

## My shortcut is missing

If the shortcut files are missing after the application has been installed, verify the following:

- For the shortcut to appear on the **Start** menu, the shortcut should be located in the `CommonPrograms` or `UserPrograms` folder.
- For the shortcut to appear on the Windows desktop, the shortcut file should be located in the `CommonDesktop` or `UserDesktop` folder.
- The **Disposition** setting for the shortcut files and folders should be set to **Virtual-integrated (layer 3)**. This allows that shortcut to be accessible on the client computer's operating system.
- The security setting on the shortcut files should not have the **Hide from folder listing** option selected. If this option is selected, the shortcut files will not be visible on the client computer.

## My shortcut is a blank window

If the desktop icon for the application is a blank Windows icon, the executable file for the application may be set to **Prevent Copying** under **Security settings** in the **File Properties** dialog box. This is normal behavior with this setting and the application functions normally.

## The text is in the wrong font

Most applications come with their own program fonts. Fonts do not appear if an application is packaged with **Inject fonts for use only by the application** selected under **Virtualization** in the **Settings** dialog box. Private fonts are only used in the application if it is launched from the Amazon WorkSpaces Application Manager Studio.

### To make private fonts available to all applications

1. In Amazon WorkSpaces Application Manager Studio, choose the **Projects** tab on the top of the project pane.
2. Under **3. Optional settings**, choose **Settings**.
3. In the **Settings** dialog box, choose **Virtualization**. Under **Fonts**, choose **Register fonts for system-wide use** and **OK**.

In addition to the **Fonts** tab, registered fonts can be found in the following locations:

- In the `fonts` directory in `SystemFonts` on the **Files** tab.
- In the following registry key on the **Registry** tab:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Fonts
```

## My folders and files are not in the application in my WorkSpace

If you package an application to an install folder that has the same name as a folder used in a WorkSpace, the folder in the WorkSpace may disappear. This can also affect files under common folders marked with Virtual-integrated (layer 3) disposition for integrated, but not merged.

### To fix an existing application package with missing folders and files

1. Use a unique name for the install folder.
2. In Amazon WorkSpaces Application Manager Studio, choose the **Files** tab on the top of the project pane.
3. Open the context (right-click) menu and then choose **Properties**.
4. In the **File Properties** dialog box, choose **Virtual-isolated (layer 4)** under **Disposition** and **OK**.

## Other Issues

The following are general issues that you might have when packaging an application.

### Issues

- [My users see a warning message about an alternate data stream \(ADS\) \(p. 65\)](#)
- [My folder, which should have files from other applications, contains only files from my application \(p. 65\)](#)

## My users see a warning message about an alternate data stream (ADS)

Users may receive warning messages about alternate data streams (ADS) when their application starts in WorkSpaces. In most cases, these messages are not a serious problem and can be ignored. In rare cases, the application may require an ADS to use a particular file. In this case, the package should exclude the file from the sandbox. Set the disposition of the file to Installed-permanent (layer 1) or Installed-temporary (layer 2).

### To exclude a file from the sandbox and set the disposition

1. In Amazon WorkSpaces Application Manager Studio, choose the **Files** tab on the top of the project pane.
2. Open the context menu (right-click) for the file and choose **Exclude**.
3. Open the context menu (right-click) for the file and choose **Properties**.
4. In the Properties dialog box, choose **Installed-permanent (layer 1)** or **Installed-temporary (layer 2)** under **Disposition** and **OK**.

## My folder, which should have files from other applications, contains only files from my application

Your folder is a root folder whose contents are only visible to the application in the package. To see the files from other applications in the unmerged folder, update the package, and configure the folder to be a merged folder. A merged folder makes files from other application visible to all applications. To change an unmerged folder into a merged folder, open the context (right-click) menu for the unmerged folder in the **Files** tab, and then choose **Merged Folder**.

## User Account Issues

The following are possible issues user account issues that you might have when packaging an application.

### Issues

- [My roaming user accounts and folder redirection does not work \(p. 66\)](#)
- [How do I create unique user accounts? \(p. 66\)](#)

## My roaming user accounts and folder redirection does not work

Amazon WorkSpaces Application Manager Studio allows the `HKEY_CURRENT_USER` registry settings and the `%appdata%` folder contents for roaming profile support. Any limitations with Windows roaming profiles still apply.

If you are using folder redirection to a network share through a mapped drive or a UNC path, set the disposition of the folder and the files to the recommended Installed-permanent (layer 1) or Installed-temporary (layer 2). The application will not start if the files or folders are set to Virtual-integrated (layer 3) or Virtual-isolated (layer 4).

For example, if the `%appdata%` folder is being redirected to `\\fileshare\user1\appdata\`, the template folder `?roamingappdata?` must only contain files and folders with Installed-permanent (layer 1) or Installed-temporary (layer 2) disposition. Redirection to another local disk supports files set to the Virtual-integrated (layer 3) or Virtual-isolated (layer 4) disposition.

### Caution

With folder redirection, the network folder must exist prior to activating the application. For example, if `%appdata%` is mapped to `n:\{username}\appdata`, the `appdata` folder must already exist on the server.

### To enable roaming profile support

1. In Amazon WorkSpaces Application Manager Studio, choose the **Project** tab on the top of the project pane.
2. Under **3. Optional settings**, choose **Settings**.
3. In the **Settings** dialog box, choose **Sandboxing, Allow Windows to roam application settings** under **Application settings**, and **OK**.

## How do I create unique user accounts?

Some applications create a special user account when you install them. Amazon WorkSpaces Application Manager Studio cannot capture user accounts because they are not transferable between operating systems. You need to create a Configurable AppEvent to replicate the user accounts and password on the host computer programmatically when Amazon WorkSpaces Application Manager Studio installs the application.

### Note

User accounts may appear as a security identifier (SID) in the registry. For example, S-1-5-18 is the local system account. For more information, go to [Well-known security identifiers in Windows operating systems](#) in the Microsoft Knowledge Base.

# Amazon WorkSpaces Application Manager Limits

---

The following table lists the limits for Amazon WorkSpaces Application Manager (Amazon WAM). Unless indicated otherwise, these limits are per region.

## Amazon WAM Limits

Resource	Limit	Comments
Application assignments	50 per user	
Packages	None	There is no limit to the number of applications you can package, but storage fees will be applied if your packages exceed 100GB.
Applications	5 GB	Applications that are larger than 5 GB cannot be packaged using Amazon WorkSpaces Application Manager (Amazon WAM).



## Document History

---

The following table describes important additions to the *Amazon WAM Administration Guide*. We also update the documentation frequently to address the feedback that you send us.

- **Latest documentation update:** April 9th, 2015

Change	Description	Date Changed
Initial release	Initial release of the <i>Amazon WAM Administration Guide</i> .	April 9th, 2015