

AppExchange Installation Guide



Last updated: July 6, 2009

Table of Contents

What is Force.com AppExchange?.....	2
About Force.com AppExchange.....	2
How does Force.com AppExchange Work?.....	3
Packaging Glossary.....	4
Who Can Use Force.com AppExchange?.....	7
Pre-Installation Inspection.....	8
Taking a Test Drive.....	8
Verifying Requirements.....	8
Determining if the App Supports Upgrades.....	8
Installing Packages.....	10
App Rollout.....	13
Review Best Practices for Use.....	13
Deploying Installed Packages.....	13
Post Installation.....	14
Managing Package Installations.....	14
Uninstalling a Package.....	15
FAQ.....	17
Installing Packages.....	17
Index.....	20

What is Force.com AppExchange?

About Force.com AppExchange

Available for: **Enterprise, Unlimited, Professional, Developer, and Group** Editions

User Permissions Needed	
To install or uninstall packages from AppExchange	"Download AppExchange Packages"
To create new AppExchange packages	"Create AppExchange Packages"
To create test drives and publish AppExchange packages to AppExchange	"Upload AppExchange Packages"

Force.com AppExchange is an application-sharing service from salesforce.com that allows you to browse, install, and share apps and components stored in packages and built for the Force.com platform. Each app is uploaded to Force.com AppExchange, where it can be listed in appropriate categories to make it easy for users to find the app they need. Users can share uploaded apps privately with a particular user or publicly with all users on AppExchange.

AppExchange supports three types of apps:

- A native app is built 100% on the Force.com platform.
- A composite app contains elements from the Force.com platform and other elements from external systems.
- A client app accesses Salesforce via the Web Services API, but does not use any other part of the Force.com platform.

The AppExchange supports the following roles throughout the life of an app:

A developer creates an app

The developer of an app is the Salesforce user who creates an app. If an app is a package, it may contain an app and its components, including custom objects, custom tabs, reports, and many other items. All apps integrate with Salesforce.

A publisher publishes the app

The publisher of an app is the Salesforce user or organization who creates a profile and one or more listings for the app on AppExchange. Publishers listings contain a link to an app they have uploaded to AppExchange, or to a third-party website. Publishers also set default license settings for managed packages.

An installer installs the app in a Salesforce organization

The installer of a package is the person who visits the AppExchange, finds a package, and installs it into a Salesforce organization.

Users use the app

Once the publisher installs the app from the AppExchange into a Salesforce organization, users in that organization can use the app.

After users install an app, they can customize it to fit the needs of their business. Therefore, the same installed app may have different custom fields, tabs, or custom s-controls in other Salesforce organizations.

The same person may perform one or more roles.

Like most Internet marketplaces, AppExchange contains detailed information about each app:

- Anyone can search for apps, and explore detailed specifications, reviews of the app from the Salesforce user community, or study ratings of the app. The publisher of the app is listed, so anyone can contact the publisher with questions.
- Users, who may also be installers or developers, can also:
 - Post reviews.
 - Click **Test Drive** to run a fully-functioning demo of the app as a read-only user in a familiar Salesforce environment.
 - Click **Get It Now** to begin the install process.
- Publishers can create public and private listings for their apps, and create or modify a publisher profile.



Note: All apps publicly listed on AppExchange must pass the salesforce.com [security review](#).

How does Force.com AppExchange Work?

The creative power behind each AppExchange app is a Salesforce user. AppExchange gives users a place to create, publish, or install apps and extensions to Salesforce. Briefly, here is how Force.com AppExchange works.

Create

Any Salesforce user initiates the process by creating a custom object, custom tab, or set of dashboards and reports that improves their business. These customers can share their customizations with other Salesforce users as an app or extension.

Publish

Any Salesforce customer can register and publish their apps on AppExchange. They can also create a demo of the functionality included in the app for customers to view before installing.

Install

Salesforce administrators can choose to install any app published to AppExchange. Installing a published app from AppExchange is safe and simple:

Browse

View the descriptions, reviews, and demos of any app from AppExchange. Choose an app that fits your needs.

Test Drive

Review a fully-functional demo of the app as a read-only user and plan your implementation.

Install

Add the app and all its components to your Salesforce environment. Deploy Immediately give your users access to the app or customize it for a select group of users.

Deploy

Immediately give your users access to the app or customize it for a select group of users.

Packaging Glossary

Available in: **Group, Professional, Enterprise, Unlimited, and Developer** Editions

Review the following terminology with Force.com AppExchange:

App

A collection of components such as tabs, reports, dashboards, and Visualforce pages that address a specific business need. Short for "application." Salesforce provides standard apps such as Sales and Call Center. You can customize the standard apps to match the way you work. In addition, you can package an app and upload it to Force.com AppExchange along with related components such as custom fields, custom tabs, and custom objects. Then, you can make the app available to other Salesforce users from AppExchange.

AppExchange Directory

A Web directory where hundreds of apps are available to Salesforce customers to review, demo, comment upon, and/or install. Developers can submit their apps for listing on [AppExchange](#) if they wish to share them with the community.

Beta, Managed Package

In the context of managed packages, an early version of a managed package that is uploaded to Force.com AppExchange for a sampling of your intended audience to test it.

Deploy

To move functionality from an inactive state to active. For example, when developing new features in the Salesforce user interface, you must select the "Deployed" option to make the functionality visible to other users.

The process by which an application or other functionality is moved from development to production.

To move metadata components from a local file system to a Salesforce organization.

For installed apps, deployment makes any custom objects in the app available to users in your organization. Before a custom object is deployed, it is only available to administrators and any users with the "Customize Application" permission.

Deprecated Component

A publisher may decide to refine the functionality in a managed package over time as the requirements evolve. This may involve redesigning some of the custom objects or fields in the managed package. Publishers cannot delete some components in a Managed - Released package, but they can deprecate a component in a later package version so that new subscribers do not receive the component, while the component continues to function for existing subscribers and API integrations. A publisher can deprecate Apex methods, classes, exceptions, enums, interfaces, or variables.

License Management Application (LMA)

A free AppExchange app that allows you to track sales leads and accounts for every user who downloads a managed package of yours from AppExchange.

License Management Organization (LMO)

The Salesforce organization that you use to track all the Salesforce users who install your package. A license management organization must have the License Management Application (LMA) installed. It automatically receives notification every time your package is installed or uninstalled so that you can easily notify users of upgrades. You can specify any

Enterprise, Unlimited, or Developer Edition organization as your license management organization. For more information, go to sites.force.com/appexchange/publisherHome.

Managed Package

A collection of application components that are posted as a unit on AppExchange, and are associated with a namespace and a License Management Organization. A package must be managed for it to be published publicly on AppExchange, and for it to support upgrades. An organization can create a single managed package that can be downloaded and installed by many different organizations. They differ from unmanaged packages in that some components are locked, allowing the managed package to be upgraded later. Unmanaged packages do not include locked components and cannot be upgraded. In addition, managed packages obfuscate certain components (like Apex) on subscribing organizations, so as to protect the intellectual property of the developer.

Managed Package Extension

Any package, component, or set of components that adds to the functionality of a managed package. An extension requires that the base managed package be installed in the organization.

Namespace Prefix

In a packaging context, a namespace prefix is a one to 15-character alphanumeric identifier that distinguishes your package and its contents from packages of other developers on AppExchange. Namespace prefixes are case-insensitive. For example, ABC and abc are not recognized as unique. Your namespace prefix must be globally unique across all Salesforce organizations. It keeps your managed package under your control exclusively.

Native App

An app that is built exclusively with setup (metadata) configuration on Force.com. Native apps do not require any external services or infrastructure.

Package

A group of Force.com components and applications that are made available to other organizations through the AppExchange. You use packages to bundle an app along with any related components so that you can upload them to Force.com AppExchange together.

Package Dependency

Created when one component references another component, permission, or preference, which must exist for the component to be valid. Components can include but are not limited to:

- Standard or custom fields
- Standard or custom objects
- Visualforce pages
- Apex scripts

Permissions and preferences can include but are not limited to:

- Divisions
- Multicurrency
- Record types

Package Installation

Incorporates the contents of a package into your Salesforce organization. A package on AppExchange can include an app, a component, or a combination of the two. After you install a package, you may need to deploy components in the package to make it generally available to the users in your organization.

Package Publication

Publishing your package makes it publicly available on Force.com AppExchange. Apps can be found under specific categories and by doing a search for keywords.

Package Registration

Registering your package allows you to access information about users who visit or download your app from the Force.com AppExchange.

Package Version

A package version is a set of components uploaded in a package. It is denoted by a version number, *majorNumber.minorNumber*, such as 2.1. Unmanaged packages are not upgradeable so each package version is simply a set of components for distribution. A package version has more significance for managed packages. Packages can exhibit different behavior for different versions. Publishers can use package versions to evolve the components in their managed packages gracefully by releasing subsequent package versions without breaking existing customer integrations using the package.

Private Sharing

The process of privately sharing a package uploaded to the AppExchange, by using the URL that you receive from Salesforce upon upload. This URL is not listed in the AppExchange. Using the unlisted URL allows you to share a package manually without making it public.

Publisher

The publisher of a package is the Salesforce user or organization that published the package on AppExchange. Publishers are required to register using their contact information after registering their package on AppExchange.

Subscriber

The subscriber of a package is a Salesforce user with an installed package in their Salesforce organization.

Test Drive

A test drive, also known as a demo, is a fully functional Salesforce organization that contains an app and any demo records added by the publisher for a particular package. It allows users on Force.com AppExchange to experience an app as a read-only user using a familiar Salesforce interface.

Unmanaged Package

An AppExchange package that cannot be upgraded or controlled by its developer. Unmanaged packages allow you to take any app components and move them "as is" to AppExchange without going through a lengthy publishing process.

Upgrading

Upgrading a package is the process of installing a newer version. Salesforce supports upgrades for managed packages that are not beta.

Uploading

Uploading a package sends it to Force.com AppExchange. Uploading your package is just the first step; all apps on AppExchange must be registered, making them privately available. The final and optional step is to publish it, making it publicly available on AppExchange.

Who Can Use Force.com AppExchange?

Anyone with a valid Salesforce username and password, including trial users, can use AppExchange to browse and test drive apps. Salesforce administrators and users with the "Customize Application" permission can install and deploy AppExchange packages into their own environments. To allow other users in your organization to use AppExchange, give them the appropriate user permission listed below:

Permission Name	Description
"Create AppExchange packages"	Create new AppExchange packages
"Download AppExchange packages"	Install or uninstall packages from AppExchange
"Upload AppExchange packages"	Create test drives and upload AppExchange packages to AppExchange

Pre-Installation Inspection

Before you begin, review the apps and determine how they fit into your current implementation of Salesforce.

Taking a Test Drive

A test drive is a new fully functional Developer Edition Salesforce environment that contains the app. Go to sites.force.com/appexchange to test drive the app before you install it.

Verifying Requirements

Every application has its own requirements. To verify that an app is compatible with your organization, click the Spec tab and review the system requirements information.

Determining if the App Supports Upgrades

Upgradeable apps are those that come from managed packages.




About Managed Packages

Available in: **Developer** Edition

Package uploads and installs are available in **Group, Professional, Enterprise, Unlimited, and Developer** Editions

A managed package is a collection of application components that are posted as a unit on AppExchange, and are associated with a namespace and a License Management Organization. A package must be managed for it to be published publicly on AppExchange, and for it to support upgrades. An organization can create a single managed package that can be downloaded and installed by many different organizations. They differ from unmanaged packages in that some components are locked, allowing the managed package to be upgraded later. Unmanaged packages do not include locked components and cannot be upgraded. In addition, managed packages obfuscate certain components (like Apex) on subscribing organizations, so as to protect the intellectual property of the developer.



Managed packages differ from unmanaged packages in many other ways. Before creating managed packages, here are a few things to consider:

- You must use a Developer Edition organization to create and work with a managed package.
- A Developer Edition organization can contain a single managed package and many unmanaged packages.
- Before converting an unmanaged package to managed, enable your organization to create managed packages. Follow the instructions in "Enabling Managed Packages for Your Organization" in the Salesforce online help.
- All packages are unmanaged unless otherwise indicated by one of the following managed icons:
 -  Managed - Beta
 -  Managed - Released
 -  Managed - Installed

- You can offer beta versions of a managed package to a small sampling of your users if you are still developing the components within the package but want to allow beta testers to install it and provide feedback. For more information on how beta versions of managed packages behave, see "Beta Versions of Managed Packages" in the Salesforce online help.
- Installers of managed packages are not able to downgrade to earlier versions.
- When you release a managed package, meaning it is uploaded with the `Managed - Released` option selected, the properties of its components change to prevent developers and installers from making harmful changes. For a list of each package component type and their properties, see "Packageable Components" in the Salesforce online help. If you do not want to offer upgrades to your package, consider keeping it unmanaged.
- Some components in a released managed package are always editable. For a list of these attributes, see "Packageable Components" in the Salesforce online help.



Tip: To prevent naming conflicts, Salesforce recommends using managed packages for all packages that contain Apex. This way, all of the Apex objects contain your namespace prefix. For example, if there is an Apex class called `MyHelloWorld` and the namespace for your organization is `OneTruCode`, the class is referenced as `OneTruCode:MyHelloWorld`.

The  `Managed - Released` or  `Managed - Installed` icon on any detail or edit page indicates that the component is included in a managed package and may not be editable. Click **Display More Information** to view which attributes you can edit. For a list of packageable components and their editable attributes, see "Packageable Components" in the Salesforce online help.

Installing Packages

Available in: **Group, Professional, Enterprise, Unlimited, and Developer** Editions

User Permissions Needed

To install packages: "Download AppExchange packages"

To install a package from AppExchange, follow this procedure from the beginning. If you received an installation link, click the link and skip to [this step](#) on page 10.



Caution: On AppExchange, the  icon indicates that an app is packaged in a managed package, making it upgradeable.

If the app is in a managed package, you cannot change some Salesforce components after you install them. For a list of editable components in a managed package, see [Packageable Components](#) in the Salesforce online help.

Managed packages also have a namespace prefix that is prepended to all uniquely named components, such as custom fields. For more information on namespace prefixes, see [Registering a Namespace Prefix](#) in the Salesforce online help.

1. Go to AppExchange by clicking **Setup** ► **View Installed Packages**.
2. Click **Visit Force.com AppExchange**.
3. Browse the Force.com AppExchange site to find the app you want to install.
4. Click **Get It Now** from the app information page.

If the listing does not have a **Get It Now** option, it may be a partner app that you can download and use outside of Salesforce. Contact the publisher for more information.

5. Enter your username and password, read the user terms, select the `I have read and agreed...` checkbox, and click **Continue**.

Select your sandbox location to install the package in your organization's sandbox, if available. This package is removed from your sandbox organization whenever you create a new sandbox copy.

6. If the package is password protected, enter the password you received from the package publisher.
7. Review the package items to ensure this is the package you want to install, and click **Next**.

If you are installing an upgrade to a managed package, review the list of new items that the upgrade will add.



Note: Some package items, such as validation rules or record types, may not display in the Package Item list but are included in the package and install with the other items. If there are no items in the Package Items list, the package may contain only minor changes not recorded in the Package Items list.

8. Review the API access that package components have been granted to ensure they are acceptable, and click **Next** if they are. If they are not, click **Cancel**. For more information, see [About API and Dynamic Apex Access in Packages](#) in the Salesforce online help.
9. When installing unmanaged packages, you can select `Resolve Naming Conflicts Automatically`. When this is selected, Salesforce changes the name of the component in the package being installed.
10. In Enterprise, Unlimited, and Developer Editions, choose a security option, and click **Next**:

Admin access only

Gives no custom object permissions to any user profile except the administrator's and any profile with the "Customize Application" permission. This option also sets field-level security to visible and editable for all fields included in the new the features in this package. After installation, if you have Enterprise, Unlimited, or Developer Edition, you can give the appropriate user permissions to your user profiles as needed.

Grant access to all users

Gives full access to custom objects included in the new package. Additionally, sets field-level security to visible and editable for all fields included in the new package.

Select security settings

Allows you to choose the usage access for all existing custom and standard profiles in your organization. You can set each profile to have full access or no access for the new package and all its components.



Note: Depending on the type of install, you may not see this step. For example, in Group and Professional Editions or if the package does not contain a custom object, Salesforce skips this step and the next, giving all users full access.

11. In Enterprise, Unlimited, and Developer Editions, if you chose **Select security settings**, select the level of access to give users in each profile and click **Next**. Standard options are:

- Full Access - Gives full access to custom objects included in the new package. Additionally, sets field-level security to visible and editable for all fields included in the new package.
- No Access - Gives no custom object permissions to any user profile except the administrator's and any profile with the "Customize Application" permission. This option also sets field-level security to visible and editable for all fields included in the new the features in this package.

You may see other options if the publisher has included settings for custom profiles. You can incorporate the settings of the publisher's custom profiles into your profiles without affecting the existing settings in your profiles. Choose the name of the profile settings in the drop-down list next to the profile you want them applied to. The current settings in that profile remain intact.

Alternatively, click **Set All** next to an access level to give this setting to all user profiles.

12. Click **Install** to install all the components in the package you have selected. See *Configuring Installed Packages* in the Salesforce online help to view the standard settings each type of component will have after you install.



Note: During installation, Salesforce automatically resolves any references to package components from the source organization. For example, if a custom link in the package refers to a document, the ID for the document will be different once you install it so Salesforce automatically alters the installed custom link to refer to the new ID in your organization. Salesforce also checks and verifies any dependencies. An installer's organization must meet all dependency requirements listed on the Show Dependencies page (with the exception of Reports) or else the installation will fail. For example, the installer's organization must have divisions enabled to install a package that references divisions. See *Understanding Dependencies* in the Salesforce online help for more information on package dependencies.

When you install a component that contains an Apex script, all unit tests for your organization are run, including the unit tests contained in the new package. If a unit test relies on a component that is initially installed as inactive, such as a workflow rule, this unit test may fail. You can select to install regardless of unit test failures.

For reasons Salesforce could prevent an install, see *Why did my install fail?* in the Salesforce online help.

13. Click **OK**.

If the package has custom objects or workflow rules, click **Deploy** and confirm when prompted to change the status of any custom objects in the package to "Deployed" and activate any workflow rules. For upgrades, this includes any new custom objects and workflow rules in the package or existing custom objects or workflow rules in the package that were not deployed previously.

14. Next, configure and deploy your package. See [Configuring Installed Packages and Deploying Installed Packages](#) in the Salesforce online help
15. If you are reinstalling a package, see [Importing Package Data](#) in the Salesforce online help to reimport the package data using the export file you received after uninstalling.
16. If you installed a managed package, click **Manage Licenses**, to assign licenses to users. See [Managing Licenses for Installed Packages](#) in the Salesforce online help.



Note: Only one person in your organization can install or upload a package at a time. Salesforce displays an error if multiple users in your organization try to install or upload packages at the same time.

App Rollout

Before you deploy the app to your users, notify them of the new functionality and educate them on how to use it. Then, deploy the new app, making it available to the users that should have permission to use it.

Review Best Practices for Use

Review the Splash Page

The developer of an app can include a splash page that users can view when they click on a custom tab. Splash pages explain how the custom object fits in a Salesforce environment and can contain training or support links. Instruct users to review the splash pages for each custom tab. Show them how to use the **Don't show me this again** link. If the publisher did not provide splash pages, create your own.

Preview the App for Select Users

Preview the app with key users in your organization. Make any suggested changes from the feedback you receive.

Announce the New App

Send an announcement to your users one week before deploying the new app. Let them know when the rollout is and how the new app benefits them. Send another announcement on the day of the deployment telling users to look for the new app.

Deploying Installed Packages

Available in: **Group, Professional, Enterprise, Unlimited, and Developer** Editions

User Permissions Needed

To deploy apps:	"Customize Application"
-----------------	-------------------------

After installing a package from Force.com AppExchange, components that are stored in folders, like reports or dashboards, are instantly available to users because of their folder settings. However, custom objects are hidden and workflow rules are inactive until you deploy them. Deploy an installed custom object and activate any workflow rule after making the necessary customizations for your organization. For tips on making customizations to installed package and components before deploying, see "Configuring Installed Packages" in the Salesforce online help. Deployed packages are available to users in your organization with the appropriate profile and page layout settings.

If you have the `Customize Application` permission you can deploy an installed package:

1. Click **Setup** ► **View Installed Packages**.
2. Select the package you want to deploy.
3. Click **Deploy**.
4. Click **Deploy** to confirm or **Cancel** to return to the package detail.

Post Installation

After installing an app from Force.com AppExchange, manage it like you manage other standard apps such as Sales or Call Center.

Managing Package Installations

Available in: **Group, Professional, Enterprise, Unlimited, and Developer** Editions

User Permissions Needed	
To uninstall packages:	"Download AppExchange Packages"
To assign licenses for a managed package:	"Manage Package Licenses"
To download or delete the export file for an uninstalled package:	"Download AppExchange Packages"

View a list of the packages you installed from Force.com AppExchange and data export files from uninstalled packages by clicking **Setup** ► **View Installed Packages**. These packages can be either managed or unmanaged.



Note: Salesforce only lists license information for managed packages. For unmanaged packages, the license-related fields, such as **Allowed Licenses**, **Used Licenses**, and **Expiration Date**, will display the value "N/A."

Using this list, you can:

- Click **Uninstall** to remove the package and all its components from your Salesforce organization. See "Uninstalling a Package" in the Salesforce online help.
- Click **Manage Licenses** to assign available licenses to users in your organization. This is only available for licensed managed packages. For more information, see "Managing Licenses for Installed Packages" in the Salesforce online help.



Note: If you purchased a site license or if the managed package is not licensed, Salesforce automatically assigns licenses to all your users and the ability to manage licenses is not available. Your users can use the package as long as they have the appropriate profile and permissions.

- Click **Configure** if the publisher has included a link to an external website with information about configuring the package.
- Click the package name to view details about this package.
- View the publisher of the package.
- View the status of the licenses for this package. Available values include:
 - Trial
 - Active
 - Suspended
 - Expired
 - Free

This field is only displayed if the package is managed and licensed.

- Track the number of licenses available (`Allowed Licenses`) and the number of licenses that are assigned to users (`Used Licenses`).
- View the date your licenses for this package are scheduled to expire.
- View the date your licenses were installed.



Note: If you have not installed a licensed managed package, the `Publisher`, `Status`, `Allowed Licenses`, `Used Licenses`, and `Expiration Date` fields do not appear.

After an uninstall, Salesforce automatically creates an export file containing the package data as well as any associated notes and attachments. When the uninstall is complete, Salesforce sends an email containing a link to the export file to the user performing the uninstall. The export file and related notes and attachments are listed below the list of installed packages. We recommend storing the file elsewhere because it is only available for a limited period of time after the uninstall completes. Using this list, you can:

- Click **Download** to open or store the export file.
- Click **Del** to delete the export file.

Uninstalling a Package

Available in: Group , Professional , Enterprise , Unlimited , and Developer Editions

User Permissions Needed	
To uninstall packages:	"Download AppExchange Packages"

You can remove any installed package, including all of its components and all data in the package. Additionally, any custom fields, links, or anything else you added to the custom app after installation are also removed.

To remove a package:

1. Click **Setup** ► **View Installed Packages**.
2. Click **Uninstall** next to the package that you want to remove.
3. Select **Yes, I want to uninstall...** and click **Uninstall**.
4. After an uninstall, Salesforce automatically creates an export file containing the package data as well as any associated notes and attachments. When the uninstall is complete, Salesforce sends an email containing a link to the export file to the user performing the uninstall. The export file and related notes and attachments are listed below the list of installed packages. We recommend storing the file elsewhere because it is only available for a limited period of time after the uninstall completes.



Tip: If you reinstall the package later and want to reimport the package data, see "Importing Package Data" in the Salesforce online help.

Notes on Uninstalling Packages

- If you are uninstalling a package that includes a custom object, all components on that custom object are also deleted. This includes custom fields, validation rules, s-controls, custom buttons and links, as well as workflow rules and approval processes.

- You cannot uninstall a package whenever any component in the package is referenced by a component that will not get included in the uninstall. For example:
 - When an installed package includes any component on a standard object that another component references, Salesforce prevents you from uninstalling the package. This means that you can install a package that includes a custom user field and build a workflow rule that gets triggered when the value of that field is a specific value. Uninstalling the package would prevent your workflow from working.
 - When you have installed two unrelated packages that each include a custom object and one custom object component references a component in the other, Salesforce prevents you from uninstalling the package. This means that you can install an expense report app that includes a custom user field and create a validation rule on another installed custom object that references that custom user field. However, uninstalling the expense report app would prevent the validation rule from working.
 - When an installed folder contains components you added after installation, Salesforce prevents you from uninstalling the package.
 - When an installed letterhead is used for an email template you added after installation, Salesforce prevents you from uninstalling the package.
- Uninstall export files contain custom app data for your package, excluding some components such as documents and formula field values.

FAQ

Review the following list of answers to frequently asked questions about installing apps from Force.com AppExchange.

Installing Packages

Can I uninstall packages that I installed from Force.com AppExchange?

Yes. All your installed packages are listed in the View Installed Packages page. You can remove any package by clicking the **Uninstall** link next to the package name. For instructions on uninstalling a package, see "Uninstalling a Package" in the Salesforce online help .

When you uninstall a Force.com AppExchange package, Salesforce automatically creates an export file containing the package data as well as any associated notes and attachments. If you choose to install the package again, you can import this data. For information on importing Force.com AppExchange package data, see "Importing Package Data" in the Salesforce online help.

Why did my uninstall fail?

Salesforce prevents you from uninstalling a package if it causes any remaining components to malfunction. For a list of instances where an uninstall could fail, see "Uninstalling a Package" in the Salesforce online help.

Salesforce automatically creates an export file containing the package data as well as any associated notes and attachments.



Tip: If you reinstall the package later and want to reimport the package data, see "Importing Package Data" in the Salesforce online help.

Who can use Force.com AppExchange?

Anyone with a valid Salesforce username and password, including trial users, can use AppExchange to browse and test drive apps. Salesforce administrators and users with the "Customize Application" permission can install and deploy AppExchange packages into their own environments. To allow other users in your organization to use AppExchange, give them the appropriate user permission listed below:

Permission Name	Description
"Create AppExchange packages"	Create new AppExchange packages
"Download AppExchange packages"	Install or uninstall packages from AppExchange
"Upload AppExchange packages"	Create test drives and upload AppExchange packages to AppExchange

Why did my install fail?

An install can fail for several reasons:

- The package you are installing includes custom objects that will cause your organization to exceed its limit of custom objects. See the Force.com AppExchange online help for a list of these limits.
- The package you are installing includes custom tabs that will cause your organization to exceed its limit of custom tabs. See the Force.com AppExchange online help for a list of these limits.
- The package you are installing contains a component requiring a unique name, such as a custom object or s-control, and a component with the same name already exists in your organization.

- The package contains features that are only available in Edition(s) you don't have.
- You already have a more recent version of the package installed.
- The developer of the package has uploaded a more recent version of the package and has deprecated the version associated with this installation URL. Contact the publisher of the package to get the most recent installation URL.
- You are trying to install an extension to a package, and you do not yet have the base package installed.
- The package you are installing requires that certain components are enabled or installed in your organization. An installer's organization must meet all dependency requirements listed on the Show Dependencies page (with the exception of Reports) or else the installation will fail. For example, the installer's organization must have divisions enabled to install a package that references divisions. For more information, see "Understanding Dependencies" in the Salesforce online help.
- The package you are installing contains an Apex script that was created by a partner who is not a certified Force.com partner and you are not authorized to run Apex in your organization. For more information, see "What is Force.com Apex Code?" in the Salesforce online help.

Can I customize Force.com AppExchange packages?

Yes, as long as the package is unmanaged. After installing an unmanaged package from AppExchange, it works just like any other package in your Salesforce organization. You can customize the folders, page layouts, profiles, and any other components.

Managed packages are not as customizable as unmanaged. To simplify the upgrade process, Salesforce locks some components in a package when it is converted from unmanaged to managed. For a list of components that are editable in a managed package, see "Packageable Components" in the Salesforce online help.

Who can use Force.com AppExchange packages?

If you are an Enterprise, Unlimited, or Developer Edition organization, you can choose which user profiles have access to the package as part of the installation process. Packages installed in Professional and Group Edition organizations are installed with "Full Access" to all user profiles. However, regardless of Edition, all custom objects are installed in "In Development" mode which hides them from all standard users. Users must have the "Customize Application" permission to view custom objects in "In Development" mode. When you are ready to roll out the package to other users, change the custom object status to "Deployed."

How can I upgrade a package after package installation?

Managed packages are completely upgradeable. Before installing a package, contact the publisher to determine if it is managed. See "Upgrading Packages" in the Salesforce online help.

How secure are the components I'm installing?

When you install a package from Force.com AppExchange, your Salesforce data is not affected. Furthermore, the components installed are just as secure as any component within Salesforce.

Be advised that components like custom links, formula fields, and web tabs can send session IDs to external services, which, in turn, could gain access to your data. While salesforce.com makes every effort to ensure that published packages meet the highest standards for security, we cannot guarantee that privately shared packages will be 100% safe. The user installing the package should examine the components within it carefully to make sure the links are going to trusted sites.

What happens to my namespace prefix when I install a package?

A namespace prefix is a globally unique identifier that you can request if you plan to create a managed package. Salesforce automatically prepends your namespace prefix, followed by two underscores ("__"), to all unique component names in your Salesforce organization. A unique package component is one that requires a name that no other component has within Salesforce, such as custom objects, custom fields, custom links, s-controls, and validation rules. All the components from a managed package that you install from another developer contain the developer's namespace prefix in your organization. However, unmanaged packages can also have a namespace prefix if they originated from an organization that contained a

managed package. When you install an unmanaged package that contains a namespace prefix, Salesforce replaces the developer's namespace prefix with yours.

Can I reinstall a Force.com AppExchange package after uninstalling it?

Yes. You can reinstall a package in the same manner that you installed it. For information on installing packages, see "Installing a Package" in the Salesforce online help.



Note: When you uninstall a Force.com AppExchange package, Salesforce automatically creates an export file containing the package data as well as any associated notes and attachments. If you choose to install the package again, you can import this data. For information on importing AppExchange package data, see [Importing Package Data](#).

Why do the custom apps, objects, and tabs in an installed package appear to not affect the limits for my Salesforce Edition

The custom apps, tabs, and objects contained in a managed package in a trial state or the managed package of a native application in any state do not count against the limits supported for your Salesforce Edition.

Index

A

AppExchange
 downloads 14
 managed packages 8

F

Force.com AppExchange
 overview 2

M

Managed packages
 about 8

P

Packages
 deploying installed Packages 13
 installations 14
 uninstalling packages 15