

SwiftKey Integration Guide

1. Introduction

This Integration Guide provides instructions to preinstall languages and test your SwiftKey installation, and help with troubleshooting of common issues.

You will need:

- SwiftKey keyboard APK
- SwiftKey factory settings APK
- Language packs

2. SwiftKey Factory Settings

The SwiftKey factory settings APK is a simple content provider that must be installed along with the SwiftKey keyboard. It allows SwiftKey to be preinstalled and provides a different user experience to a version of SwiftKey downloaded from Google Play. The factory settings APK is not visible to users on their device home screen.

3. Downloading Language Packs

Support for different languages in SwiftKey is provided via individual 'language packs'. One language pack corresponds to the particular variant of a language used in a specific geographical locale, so UK English is 'en_GB' and US English is 'en_US'. Many languages will only have one language pack available, eg 'fi_FI' for Finnish.

If you need languages to be available on the device (ie without downloading) the first time a user interacts with SwiftKey, you must preinstall these on the device. If you do not do this, the user will have to select the desired languages and download the language packs manually.

Downloading languages for preinstallation:

Language packs can be downloaded from the following link. Make sure you download the language packs that match the correct version of SwiftKey by going to the tab that matches the version of SwiftKey you are installing:

<https://preinstall-languages.api.swiftkey.com/>

Select the languages that you want to preinstall, and click 'Download selected language packs'. This will download a file called languagePacks.zip which contains your chosen language packs as a set of zip files, and a metadata file called *lpsdata.json*.

By default SwiftKey assumes that language packs will be preinstalled in `/system/com.touchtype/` and this is also the default file location specified on the language pack download page under the heading 'Prefix to archive URL'. If you want to use a different location, please ask SwiftKey for a new Factory Settings APK including the correct path. You can then use the 'Prefix to archive URL' field to specify the new path in the generated json file.

Updating Languages:

Please note that whenever you want to preinstall a new language or a newer version of an existing language, you must download ALL of your preinstalled languages and the metadata file `lpsdata.json` again.

4. Installing and Testing SwiftKey

This section describes where SwiftKey files should be installed on a device's file system, and provides instructions for making a manual installation via the Android Debug Bridge for testing purposes. The location of the files should usually be the same whether you are making a production ROM, or manually installing files for test purposes, although you may have a specific location within the system partition you would prefer to use on your production devices.

Important note:

- We strongly recommend that all APKs and languages be installed in the system partition
- The SwiftKey factory settings APK must be installed before the SwiftKey keyboard APK

Please note that if the SwiftKey factory settings APK is installed in the data partition and not the system partition, then the user will be able to uninstall it. This means that the custom configuration may not be applied.

Before you start:

You will need to have downloaded the languages that you wish to pre install - see Section 3. You will need to have informed SwiftKey of any specific languages you would like to be enabled for a particular locale.

Please ensure that:

- You have the correct SwiftKey APK for the Android version of your device.
- You have the correct SwiftKey apk for the architecture of your device - it will be either *armeabi-v7a* or *arm64-v8a*.
- You have administrator permission levels on the device in order to install SwiftKey in the `/system/app` directory.

- You have 'read and write' permissions for the language zip files on the device. Installation may fail if they do not have read and write permissions for all users after they have been copied to the device.
- The device has at least 3MB available for each preinstalled language pack.

Installation Instructions

1. Install the SwiftKey factory settings APK first: open a command prompt and copy the SwiftKey factory settings APK to the device `/system/app` directory using the following command, substituting '`<APK_name>`' for the name of your specific factory settings APK:

```
adb push <APK_name>.apk /system/app
```

2. Install the SwiftKey APK next: at the command prompt copy the SwiftKey APK to the device `/system/app` directory using the following command:

```
adb push <APK_name>.apk /system/app
```

3. Unzip the bundle zip file (`languagePacks.zip`) to your local machine, and copy the full contents to the `/system/com.touchtype` directory on the device. This includes the language packs '*.zip' files and the metadata file `lpsdata.json`.

```
unzip -d languagePacks languagePacks.zip  
adb shell mkdir /system/com.touchtype  
adb push languagePacks/ /system/com.touchtype
```

4. Enable SwiftKey as a keyboard by running the following command:

```
adb shell ime enable com.touchtype.swiftkey/com.touchtype.KeyboardService
```

5. Set SwiftKey to be the default keyboard by running the following command:

```
adb shell ime set com.touchtype.swiftkey/com.touchtype.KeyboardService
```

Testing SwiftKey

It is vital that you conduct your own testing of the SwiftKey keyboard. Please follow these steps to confirm you have integrated SwiftKey successfully:

- 1) On first boot of the device, SwiftKey will be the default keyboard. Note that SwiftKey does not show predictions in certain field types, such as password fields, so when the keyboard is first

used (for example to enter WIFI details or register a Google account) predictions may not be shown.

- 2) The SwiftKey installer will not be shown and there will be no notifications about running the SwiftKey installer.
- 3) Open an application with a text field, such as a text messaging application. The SwiftKey Keyboard will appear and predictions will be shown in the correct language based on the device locale (provided that the locale matches one of the installed languages).
- 4) The SwiftKey Onboarding will appear when you leave a prediction field for the first time, outside of the device setup wizard.
- 5) Go to the SwiftKey settings, choose Languages, and download and enable an additional language (you will need an Internet connection). Open an application with a text field and predictions will be available in the new language.
- 6) All the changes you made to SwiftKey's default settings will have been applied.

5. How do I report a problem to SwiftKey?

SwiftKey requires the following information in order to investigate an issue:

- Brief description of issue
- Clear and precise steps to reproduce
- Log files
- SwiftKey version(s)
- Android version(s)
- Device firmware/ROM version
- Application(s) that the issue has been reproduced in
- Version of application(s)
- If the issue is visible in the UI, a video or screenshot

Please also share any upcoming release dates and if possible send us a device with root permissions to help us reproduce and fix the issue promptly.

6. FAQ

Videos and articles explaining how various features work for the end user can be found on the [Help Section](#) on the SwiftKey website.

- *What are the minimum requirements to run SwiftKey?*

Hardware

- 1 GHz CPU
- 512 MB RAM
- 512 MB storage

Operating System

- Android 5.0

- *What is the difference between preinstalled SwiftKey and the version available via Google Play?*

The preinstalled SwiftKey APK is exactly the APK distributed via Google Play - the behavioral differences are triggered by the Factory Settings APK which accompanies preinstalled SwiftKey.

If a user downloads SwiftKey from Google Play they will need to go through the SwiftKey installer process, and will require an Internet connection to download their locale language. When the SwiftKey keyboard is preinstalled along with the SwiftKey Factory Settings APK, the end user will see the SwiftKey onboarding feature instead of the SwiftKey installer and their locale language will be already set up and ready to use.

- *Why should I preinstall languages?*

Preinstalling a language that matches the locale for a given device, and/or specifying a series of locale-specific mappings, enables SwiftKey automatically to enable the relevant language(s) so that predictions will be available immediately. If the locale language is not preinstalled, and there is no Internet connection to download the relevant language pack, SwiftKey will only offer predictions in English US.

- *Why is the language list empty?*

There are a number of possible reasons:

1. Language packs have been installed in the wrong folder. The zip files containing preinstalled languages must be installed in the correct folder: either the default `/system/com.touchtype/` or an alternative location you have asked SwiftKey to specify in the Factory Settings APK.
2. Incorrect permissions. All language pack zip files must have read and write permissions for all users.
3. Corrupt language packs. Zip files may have become corrupted during the pre-installation process.
4. Missing meta data file. The language packs and the `lpsdata.json` file must both be installed in the correct language preinstall folder.

5. Out of date *lpsdata.json* file or language packs. Whenever you want to preinstall a new language or a newer version of an existing language, you must download ALL of your preinstalled languages and the metadata file *lpsdata.json* again.