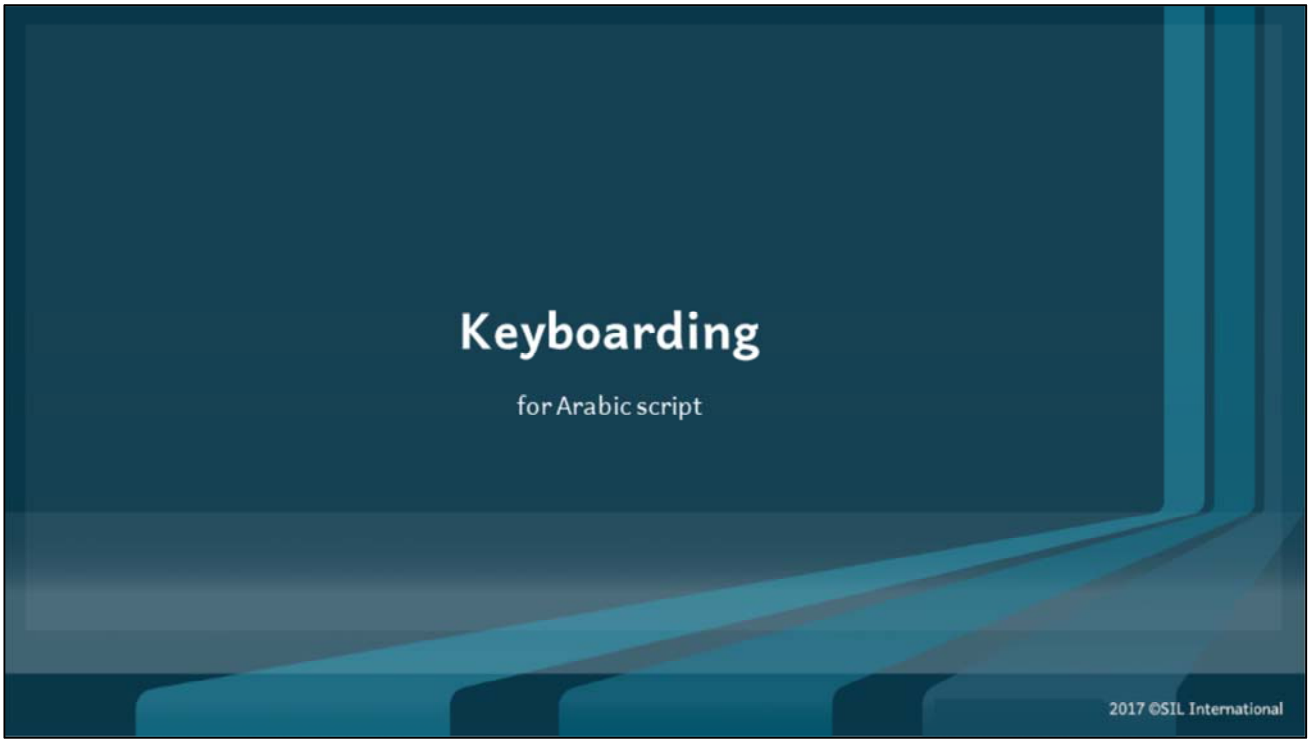
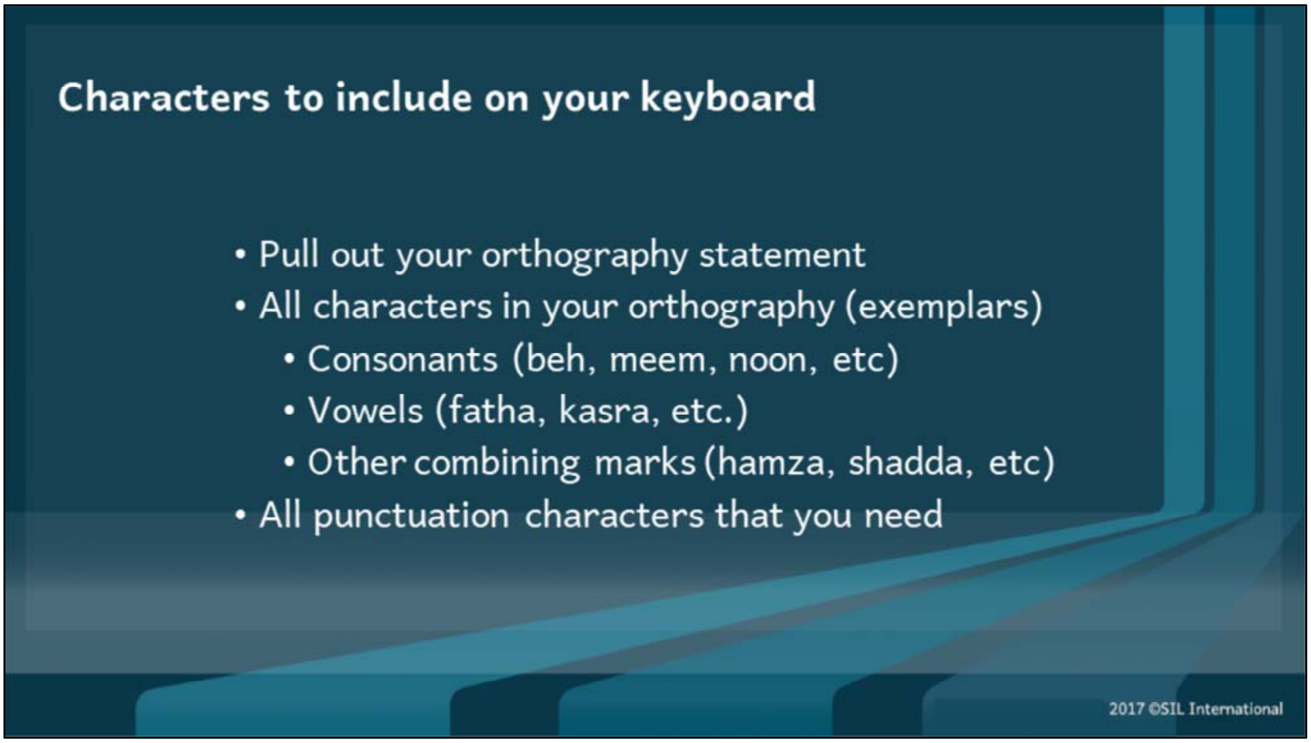


Keyboarding

for Arabic script

2017 ©SIL International

This session will be looking at what characters to include in your keyboard, how to decide on a layout, and reviewing various keyboarding technologies. Most of the time will be spent on looking at Keyman.



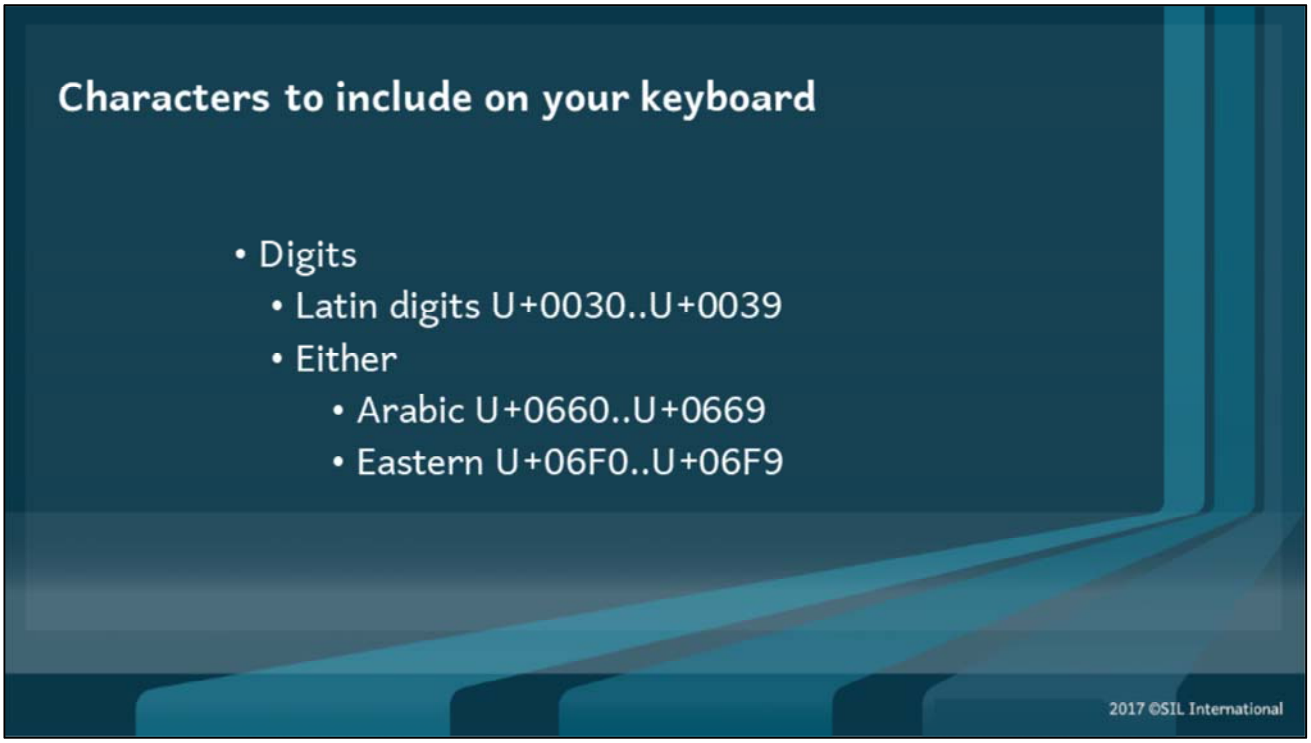
Characters to include on your keyboard

- Pull out your orthography statement
- All characters in your orthography (exemplars)
  - Consonants (beh, meem, noon, etc)
  - Vowels (fatha, kasra, etc.)
  - Other combining marks (hamza, shadda, etc)
- All punctuation characters that you need

2017 ©SIL International

First of all, we hope you have an orthography statement. If you pull that out it should list all the characters you will need. Orthography statements do not tend to include the punctuation characters (although maybe they should!).

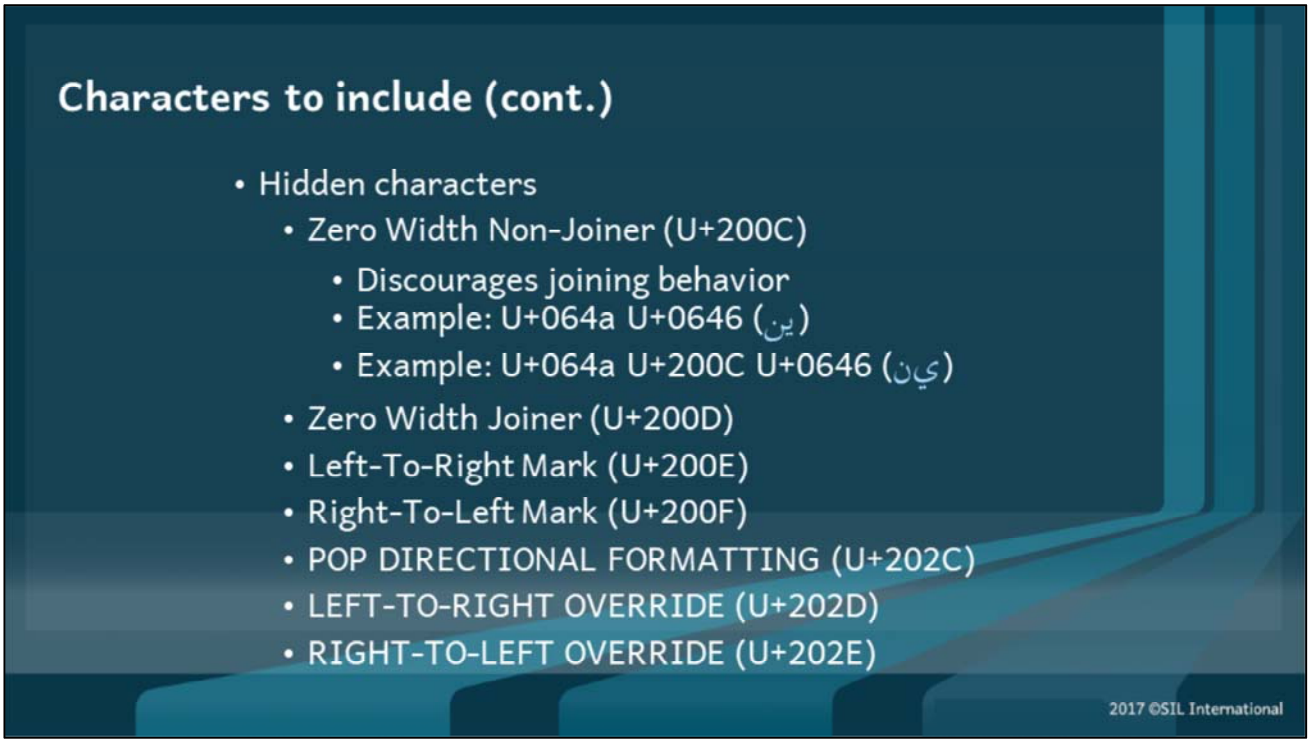
List all of the punctuation characters you need.



Characters to include on your keyboard

- Digits
  - Latin digits U+0030..U+0039
  - Either
    - Arabic U+0660..U+0669
    - Eastern U+06F0..U+06F9

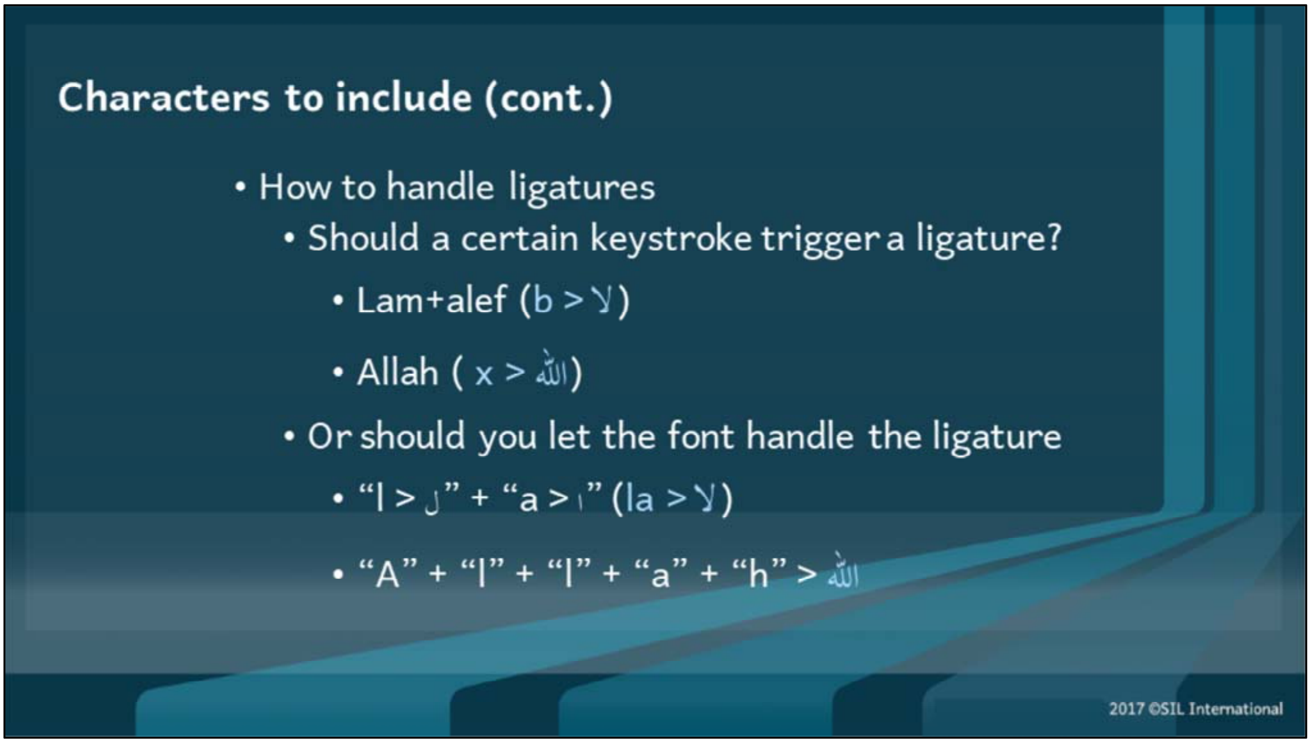
You might consider including both “Latin” digits and the Arabic digits that you require. Some applications, such as Paratext require using Latin digits for checking purposes (in chapter and verse numbers). When you export the project you can convert all digits to the required digits.



Characters to include (cont.)

- Hidden characters
  - Zero Width Non-Joiner (U+200C)
    - Discourages joining behavior
    - Example: U+064a U+0646 (ين)
    - Example: U+064a U+200C U+0646 (ي‌ن)
  - Zero Width Joiner (U+200D)
  - Left-To-Right Mark (U+200E)
  - Right-To-Left Mark (U+200F)
  - POP DIRECTIONAL FORMATTING (U+202C)
  - LEFT-TO-RIGHT OVERRIDE (U+202D)
  - RIGHT-TO-LEFT OVERRIDE (U+202E)

You should consider whether you want any hidden characters on your keyboard.

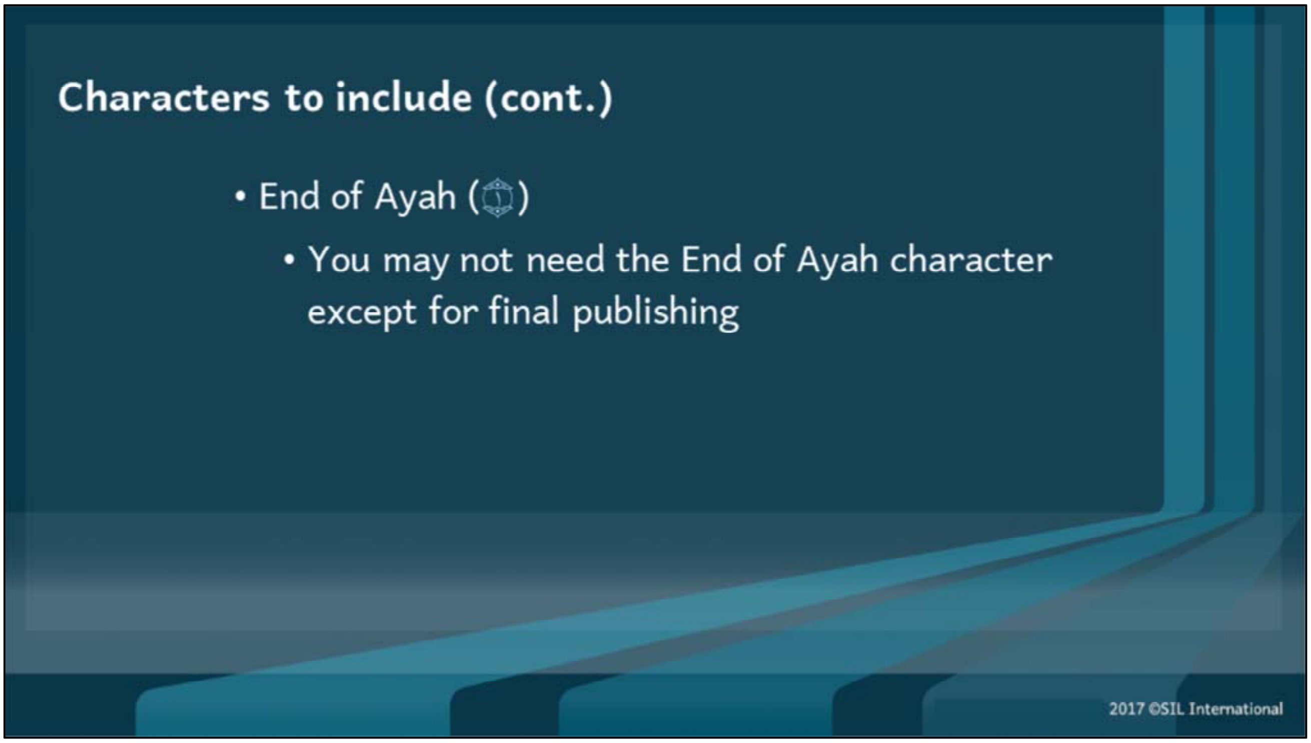


Characters to include (cont.)

- How to handle ligatures
  - Should a certain keystroke trigger a ligature?
    - Lam+alef (b > لا)
    - Allah (x > ﷲ)
  - Or should you let the font handle the ligature
    - “l > ل” + “a > ا” (la > لا)
    - “A” + “l” + “l” + “a” + “h” > ﷲ

2017 ©SIL International

You will need to think about how you want to type ligatures. Do you want one keystroke to trigger the ligature or do you want a sequence of characters (keystrokes) to trigger the ligature?

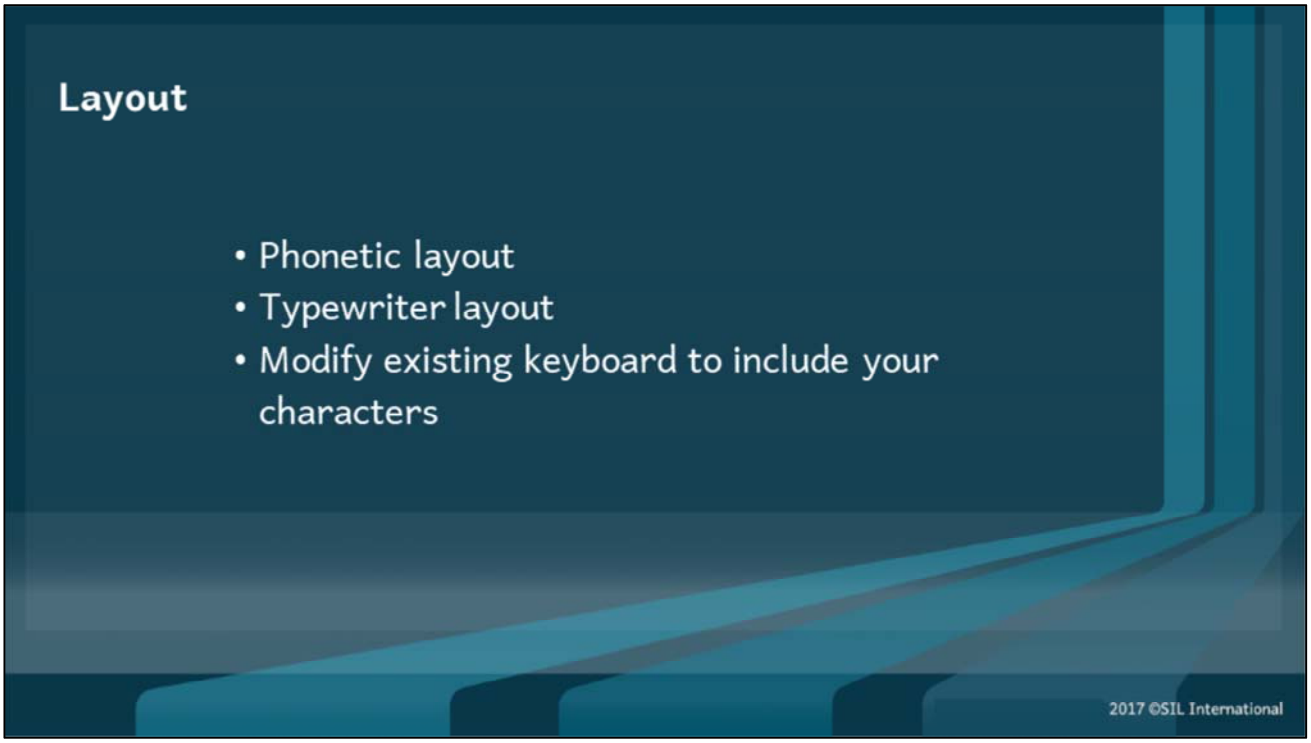


Characters to include (cont.)

- End of Ayah (۝١)
  - You may not need the End of Ayah character except for final publishing

2017 ©SIL International

You will need to think about how you want to type ligatures. Do you want one keystroke to trigger the ligature or do you want a sequence of characters (keystrokes) to trigger the ligature?



Layout

- Phonetic layout
- Typewriter layout
- Modify existing keyboard to include your characters

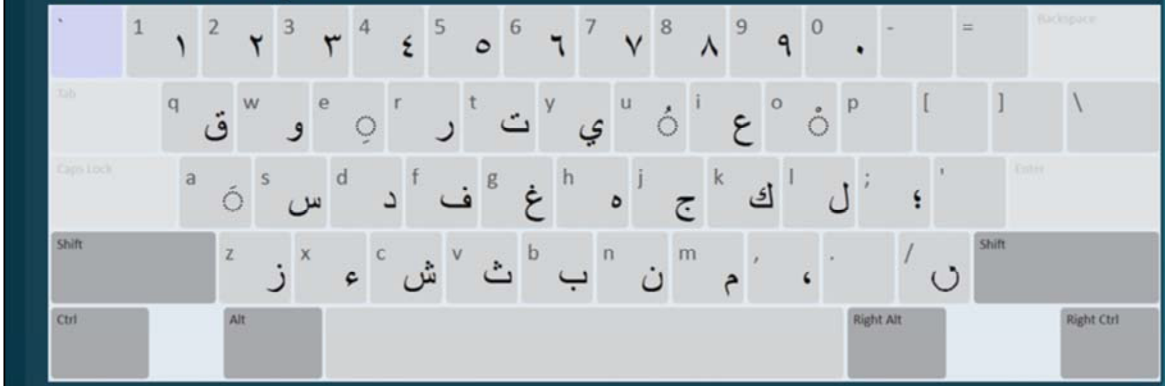
2017 ©SIL International

You will need to decide what kind of layout you want. Is this keyboard primarily for you or for the user community? If the keyboard is for you, you might like it to be phonetic as that might be an easier way for you to type.

If the keyboard is intended for mother tongue speakers, then you may want to use an Arabic typewriter layout if that is what they are used to.

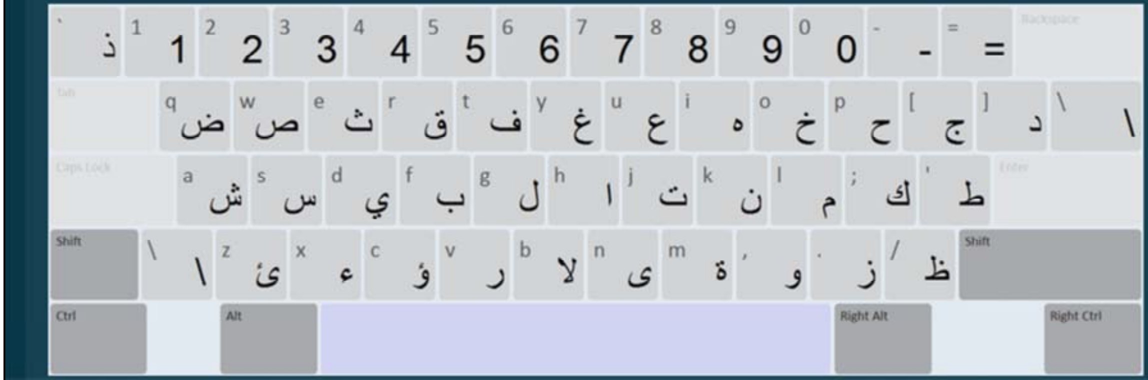
For either of these two options you can try to find an existing keyboard that you can modify. That will make it easier than creating it from scratch.

Phonetic Layout

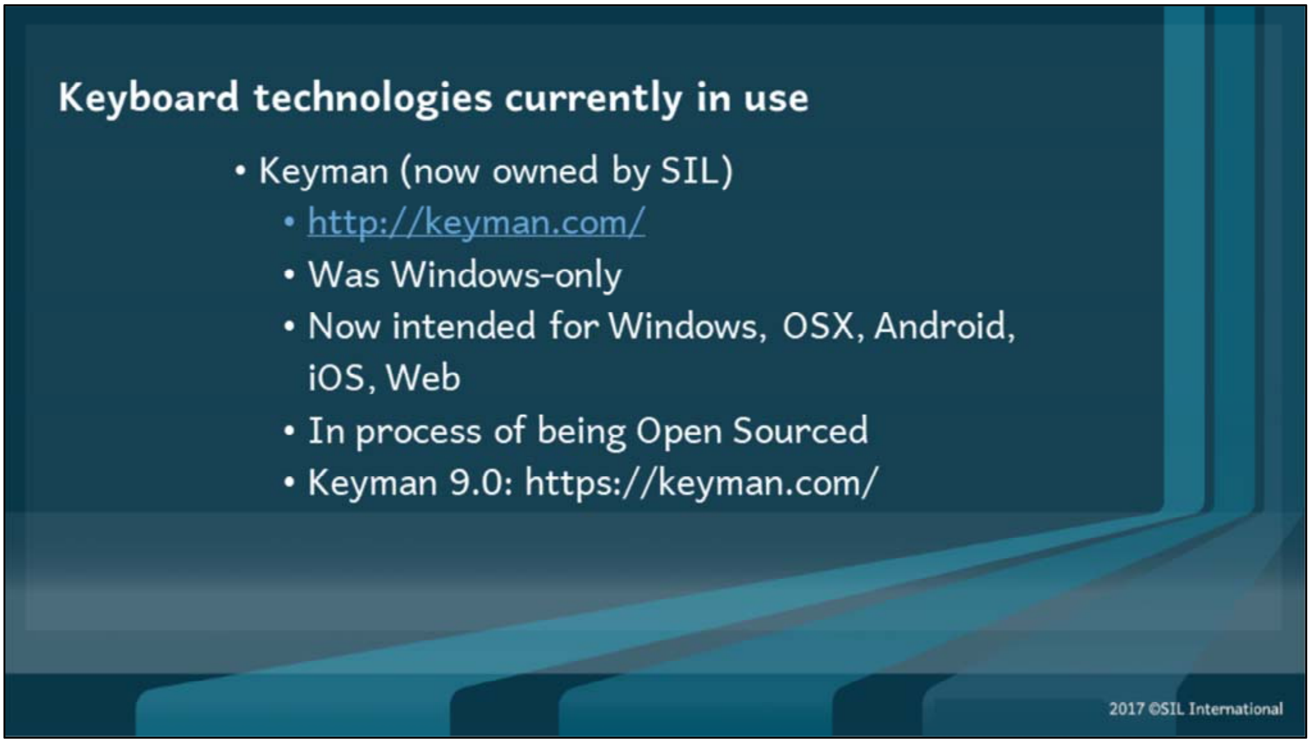


- Make it phonetic
  - b = beh, l = lam, m = meem, n = noon, etc
  - Easier for expats

Typewriter Layout



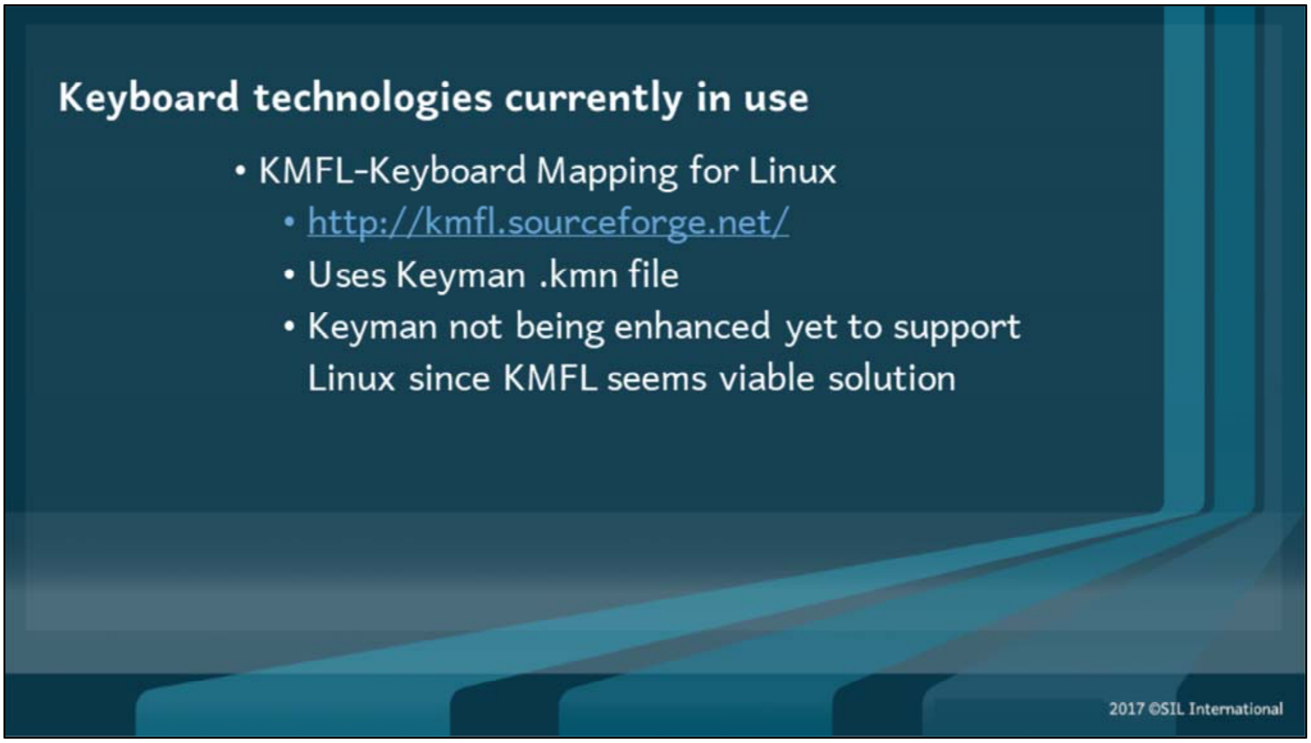
- Arabic typewriter layout
  - Easier for mother tongue speakers



Keyboard technologies currently in use

- Keyman (now owned by SIL)
  - http://keyman.com/
  - Was Windows-only
  - Now intended for Windows, OSX, Android, iOS, Web
  - In process of being Open Sourced
  - Keyman 9.0: https://keyman.com/

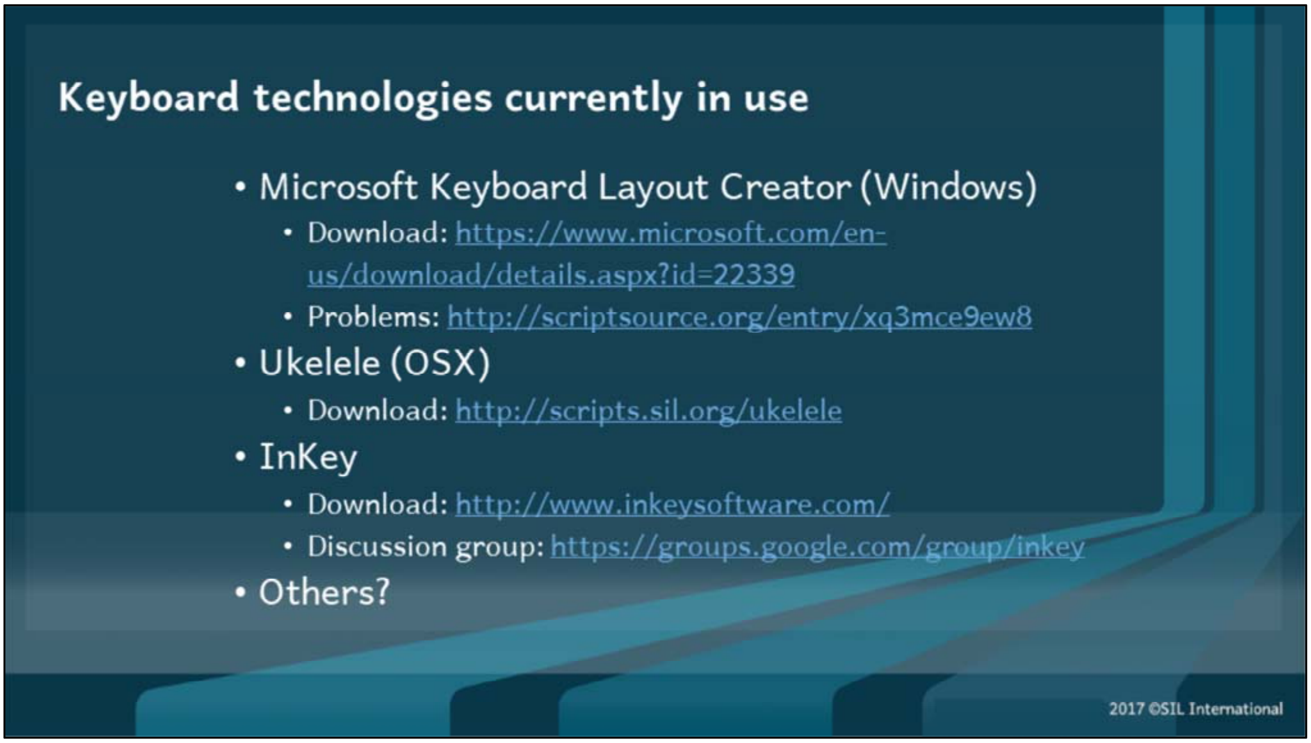
Keyman is a powerful tool from Keyman for creating keyboard input methods. It allows for more complex input processing than MSKLC and so meets a wider range of needs. It does not use the Windows keyboard driver format, so a separate client needs to be installed on each computer using the keyboard. There are many existing Keyman keyboards available. More information on Keyman can be found on SIL's NRSI site. There are now other keyboarding systems which can work with Keyman .kmn source files.



Keyboard technologies currently in use

- KMFL-Keyboard Mapping for Linux
  - http://kmfl.sourceforge.net/
  - Uses Keyman .kmn file
  - Keyman not being enhanced yet to support Linux since KMFL seems viable solution

KMFL is an open source keyboarding system for Linux systems, particularly Ubuntu, which is compatible with Keyman 7 (or earlier) .kmn source files, so brings the power of Keyman keyboarding to a Linux environment. Although designed for Linux/Unix systems in general, full downloads and instructions are currently only available for Ubuntu.



Keyboard technologies currently in use

- Microsoft Keyboard Layout Creator (Windows)
  - Download: https://www.microsoft.com/en-us/download/details.aspx?id=22339
  - Problems: http://scriptsource.org/entry/xq3mce9ew8
- Ukelele (OSX)
  - Download: http://scripts.sil.org/ukelele
- InKey
  - Download: http://www.inkeysoftware.com/
  - Discussion group: https://groups.google.com/group/inkey
- Others?

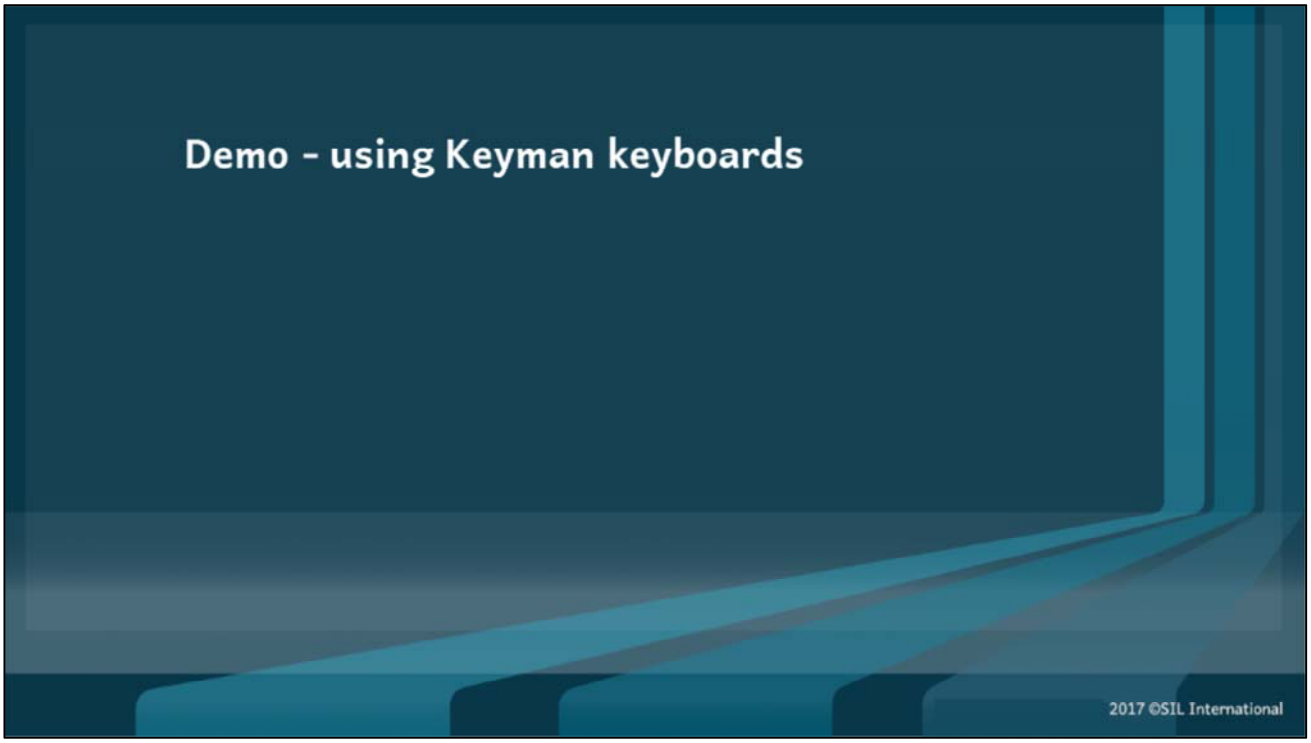
2017 ©SIL International

MSKLC is a freeware Microsoft tool for creating new keyboard layouts using the Windows-native keyboard file format, and so gives seamless integration with Windows systems. It can use an existing Windows keyboard as a starting point. However, it is limited in its ability to cope with more complex scripts.

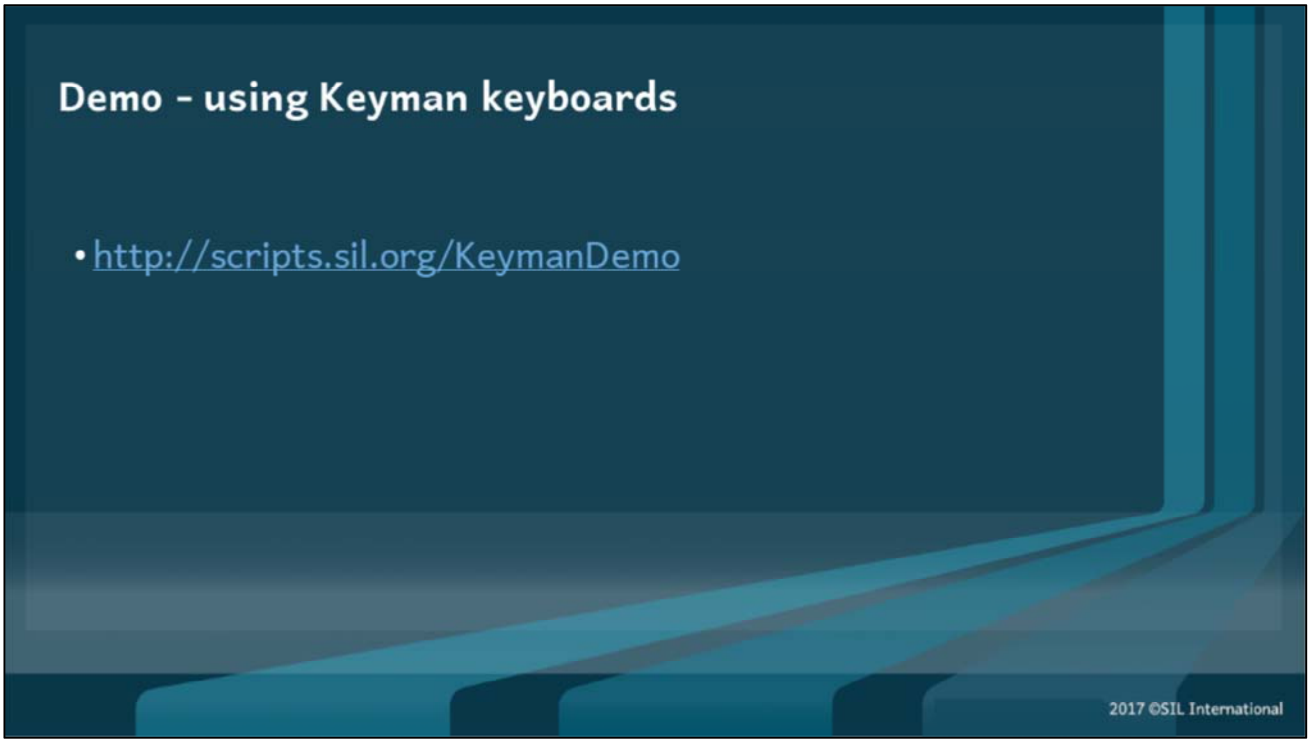
Ukelele is a freeware keyboard layout editor for OS X which provides a graphical interface for .keylayout files (the standard keyboard format for OS X).

Inkey is based on Autohotkey, an open source macro-creation and automation tool. The language used by InKey (and Autohotkey underneath) provides great flexibility and allows complex keyboard behaviours to be programmed. The separate open source Inkey Keyboard Creator is available to facilitate the initial creation of the keyboard, including an option to import Keyman .kmn files. More complex tasks have to be done by editing the Inkey .ahk source file directly, after which the keyboard creator cannot be used to make further changes.

Inkey is currently under private beta release, but the authors have indicated that the next version will be released under a free, open source license.

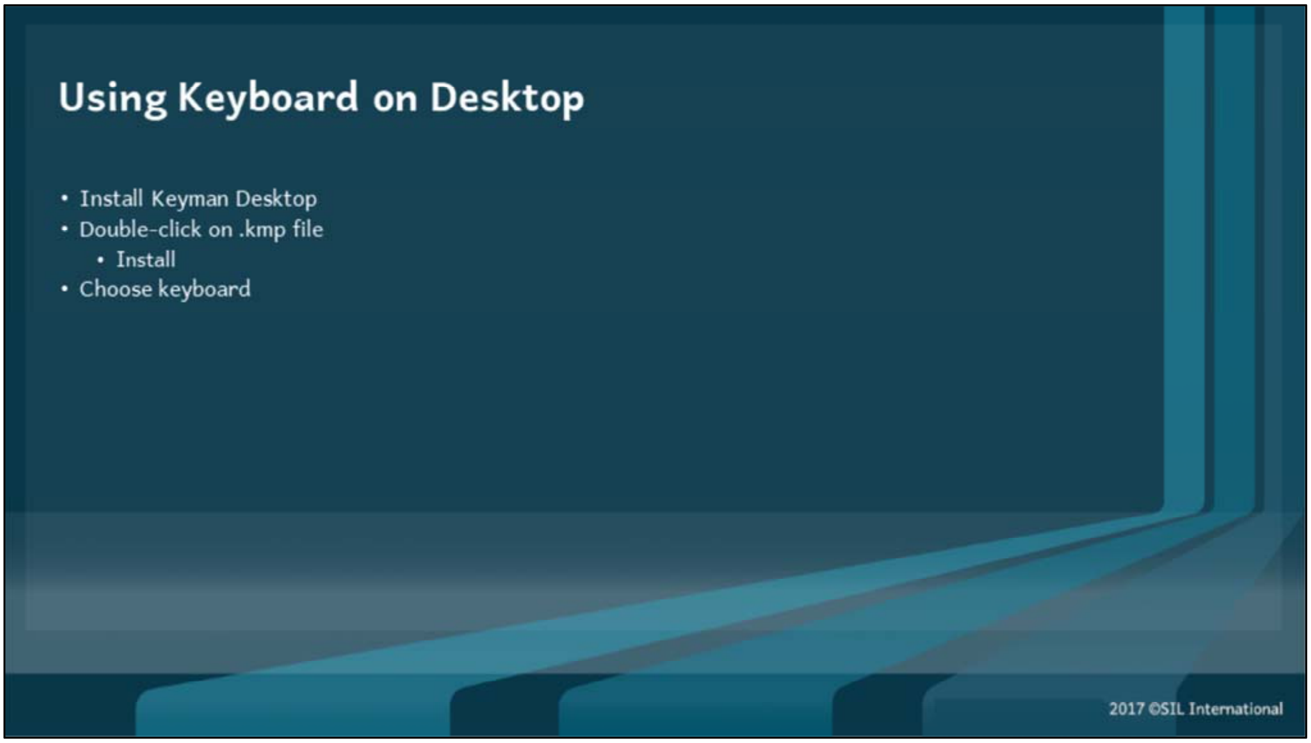


Demo - using Keyman keyboards



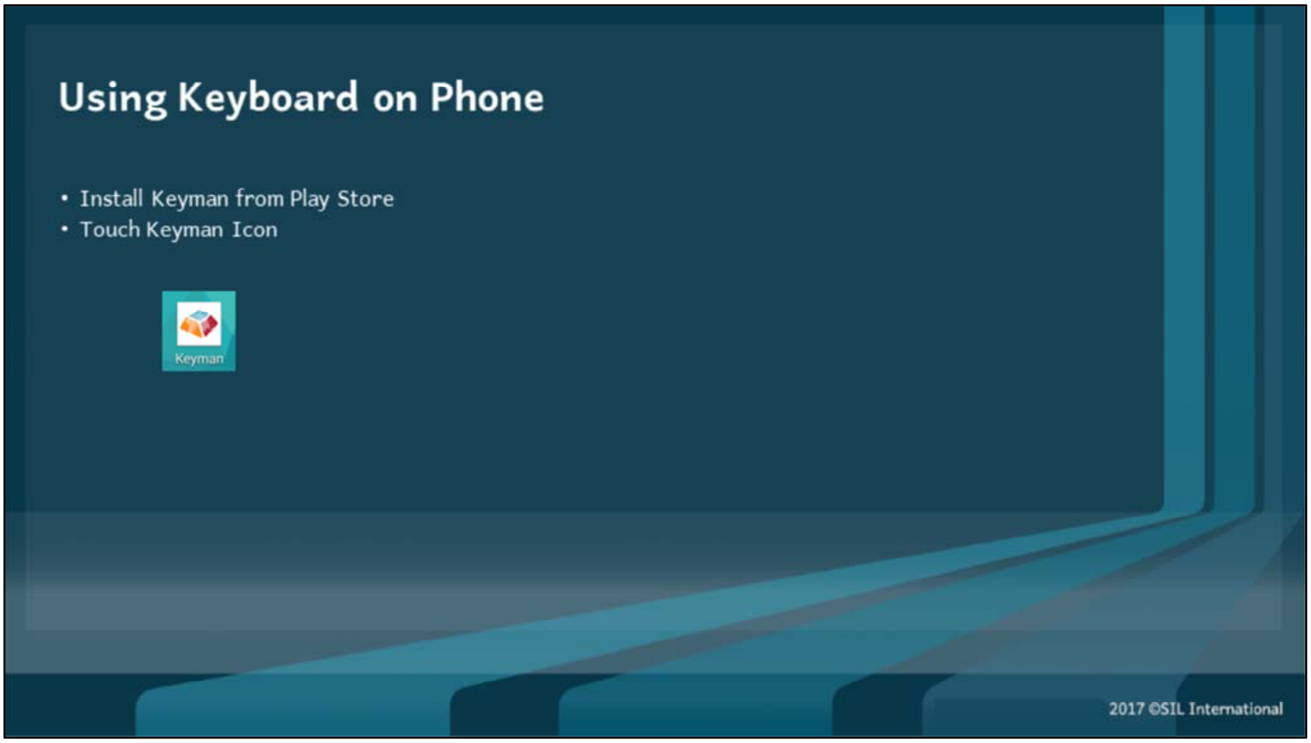
Demo - using Keyman keyboards

- http://scripts.sil.org/KeymanDemo



Using Keyboard on Desktop

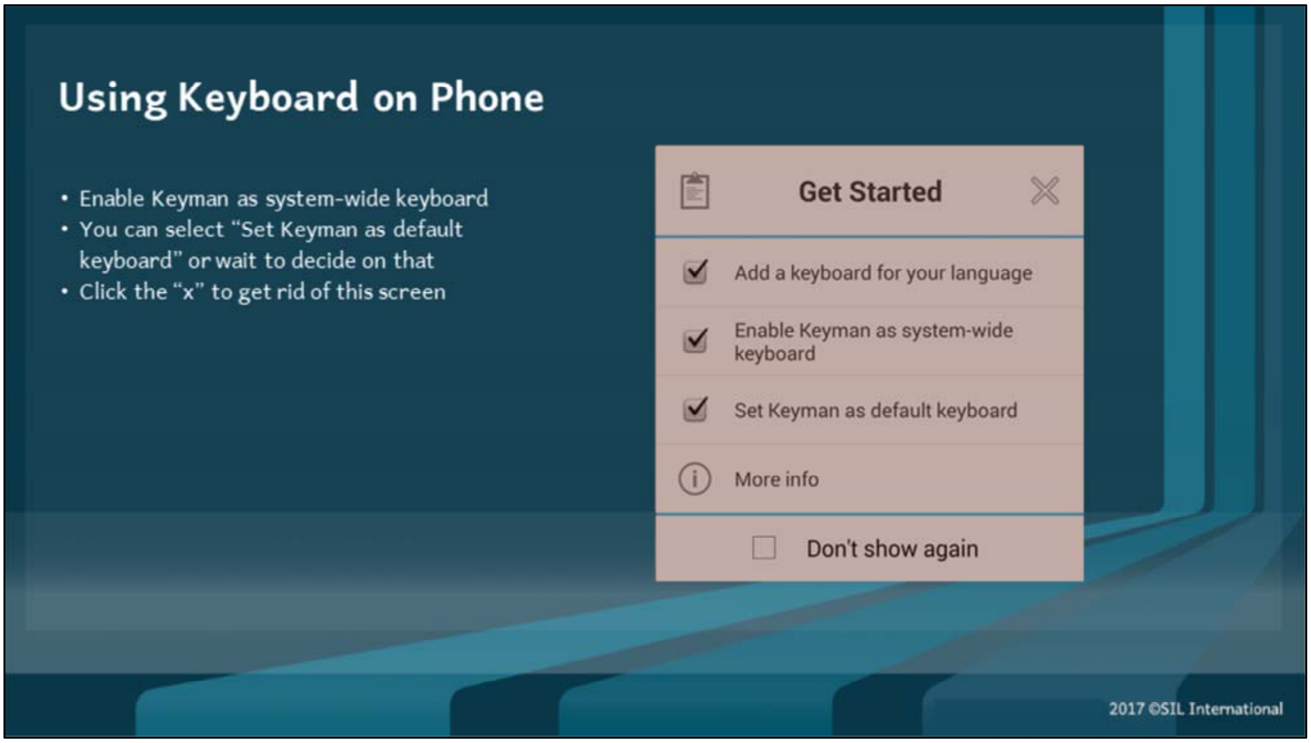
- Install Keyman Desktop
- Double-click on .kmp file
  - Install
- Choose keyboard



Using Keyboard on Phone

- Install Keyman from Play Store
- Touch Keyman Icon

When you touch the Keyman Icon it opens the Keyman app. You can type in this application by selecting the keyboard through the little “world” icon.



Using Keyboard on Phone

- Enable Keyman as system-wide keyboard
- You can select “Set Keyman as default keyboard” or wait to decide on that
- Click the “x” to get rid of this screen

Get Started ✕

☑ Add a keyboard for your language

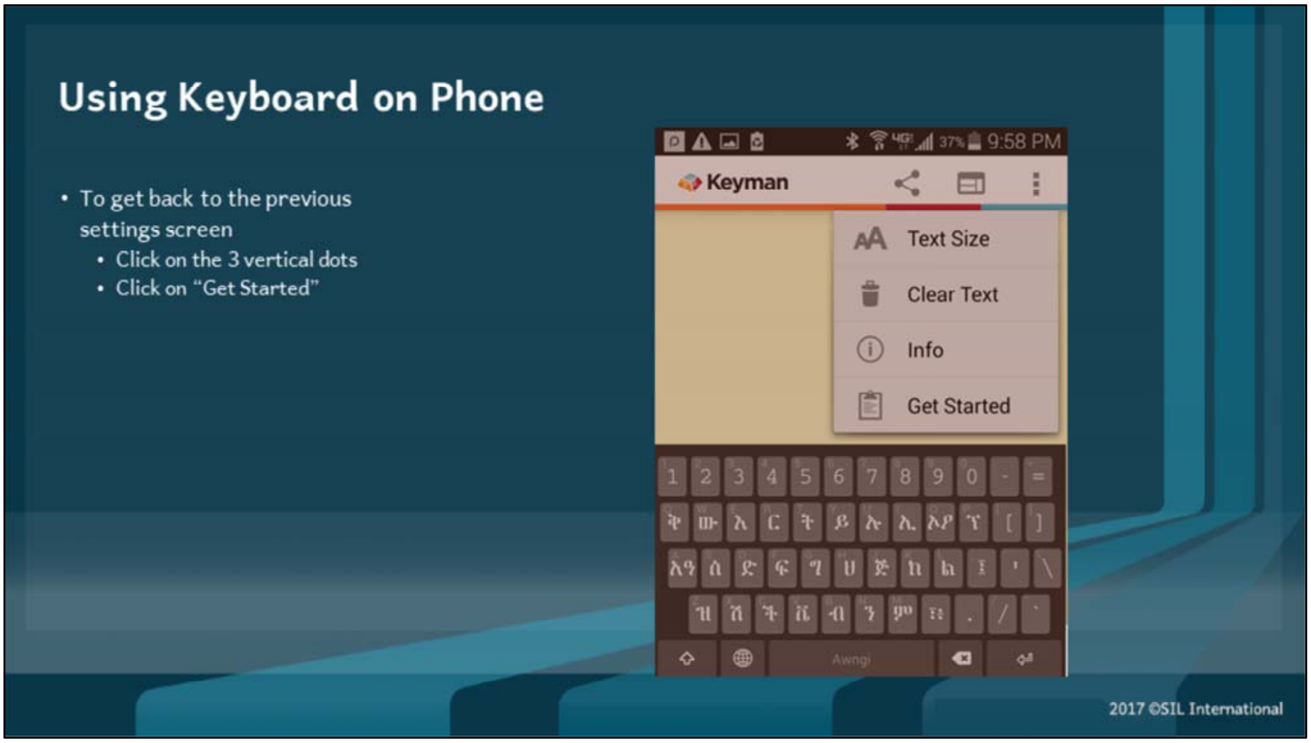
☑ Enable Keyman as system-wide keyboard

☑ Set Keyman as default keyboard

ⓘ More info

□ Don't show again

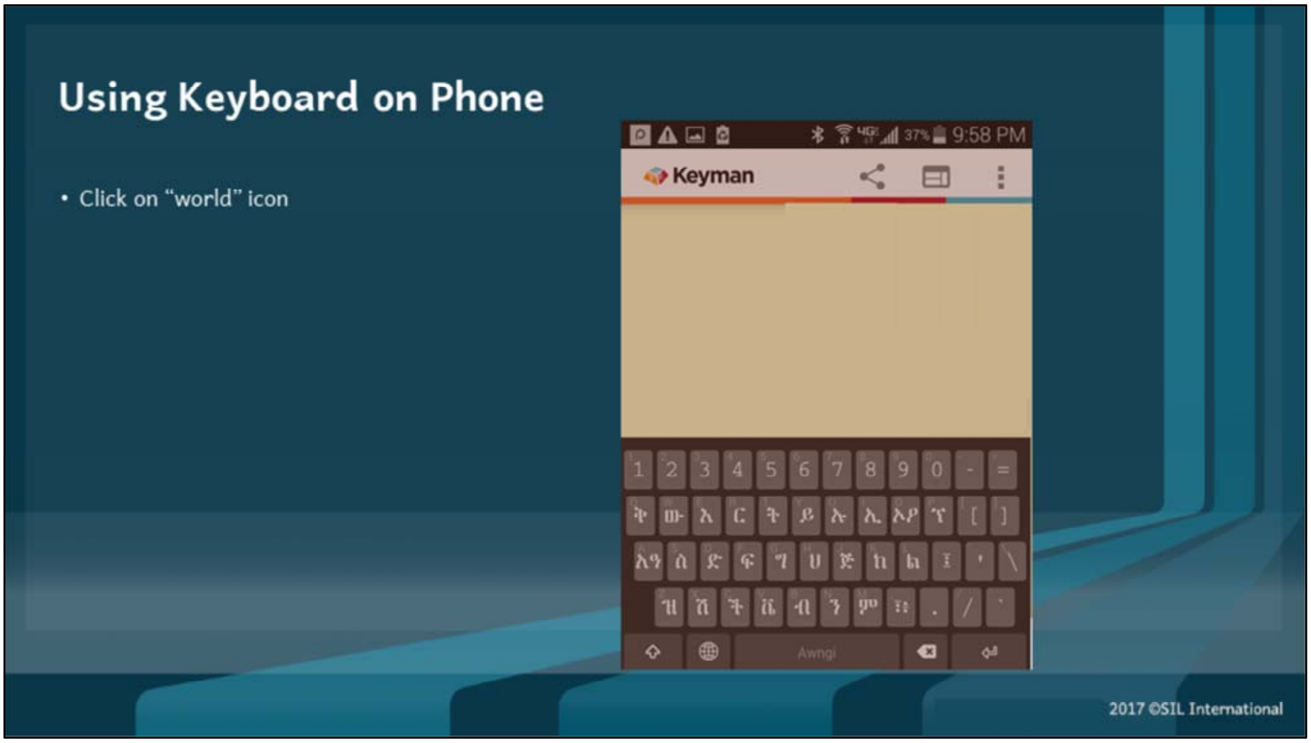
If you decide to wait to “Set Keyman as default keyboard” you can get to it another way.



Using Keyboard on Phone

- To get back to the previous settings screen
  - Click on the 3 vertical dots
  - Click on "Get Started"

You can click on the 3 vertical dots and then the "Get Started" and it will show the Keyman choices again.

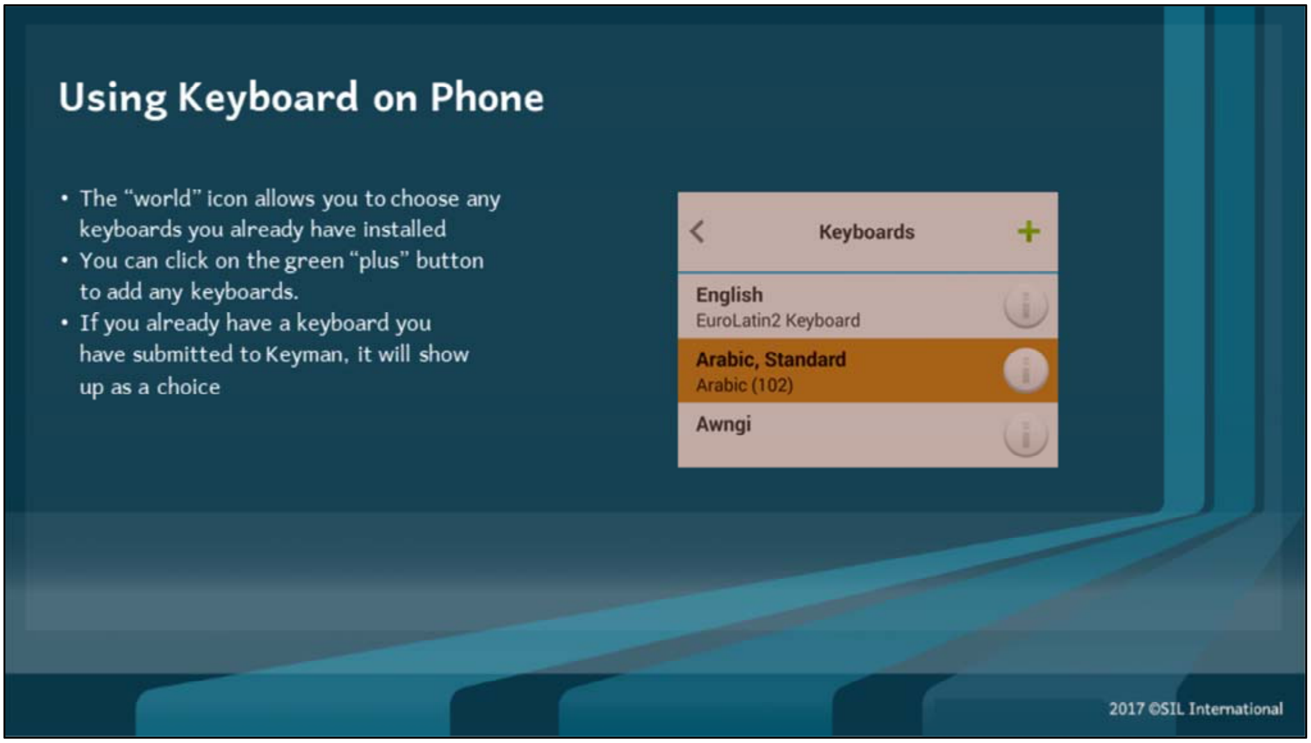


Using Keyboard on Phone

- Click on “world” icon

2017 ©SIL International

We’ve just opened the Keyman app. You can type in this application by selecting the keyboard through the little “world” icon.



Using Keyboard on Phone

- The "world" icon allows you to choose any keyboards you already have installed
- You can click on the green "plus" button to add any keyboards.
- If you already have a keyboard you have submitted to Keyman, it will show up as a choice

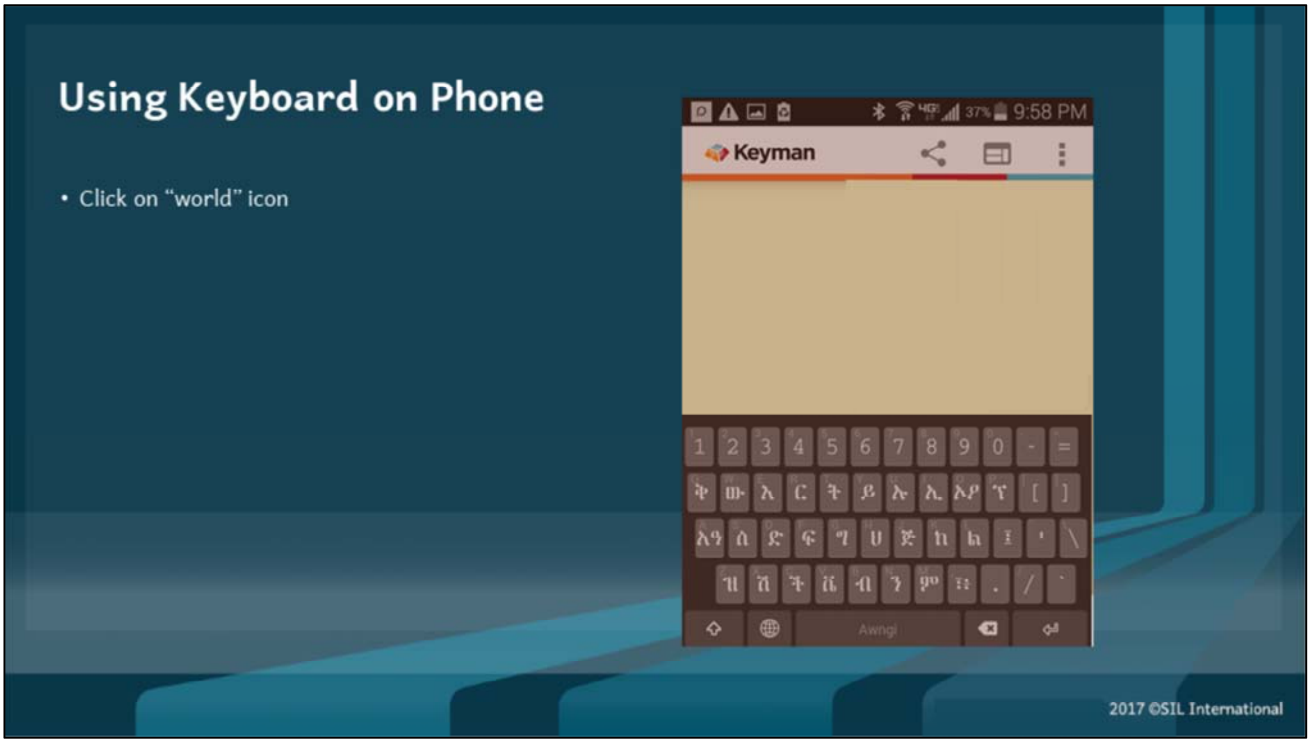
English
EuroLatin2 Keyboard

Arabic, Standard
Arabic (102)

Awngi

2017 ©SIL International

Now you can type in the Keyman app. Then you will need to copy and paste into wherever you want the text.
This is probably not exactly what you will want. In that case, you will want to set Keyman as your default keyboard.

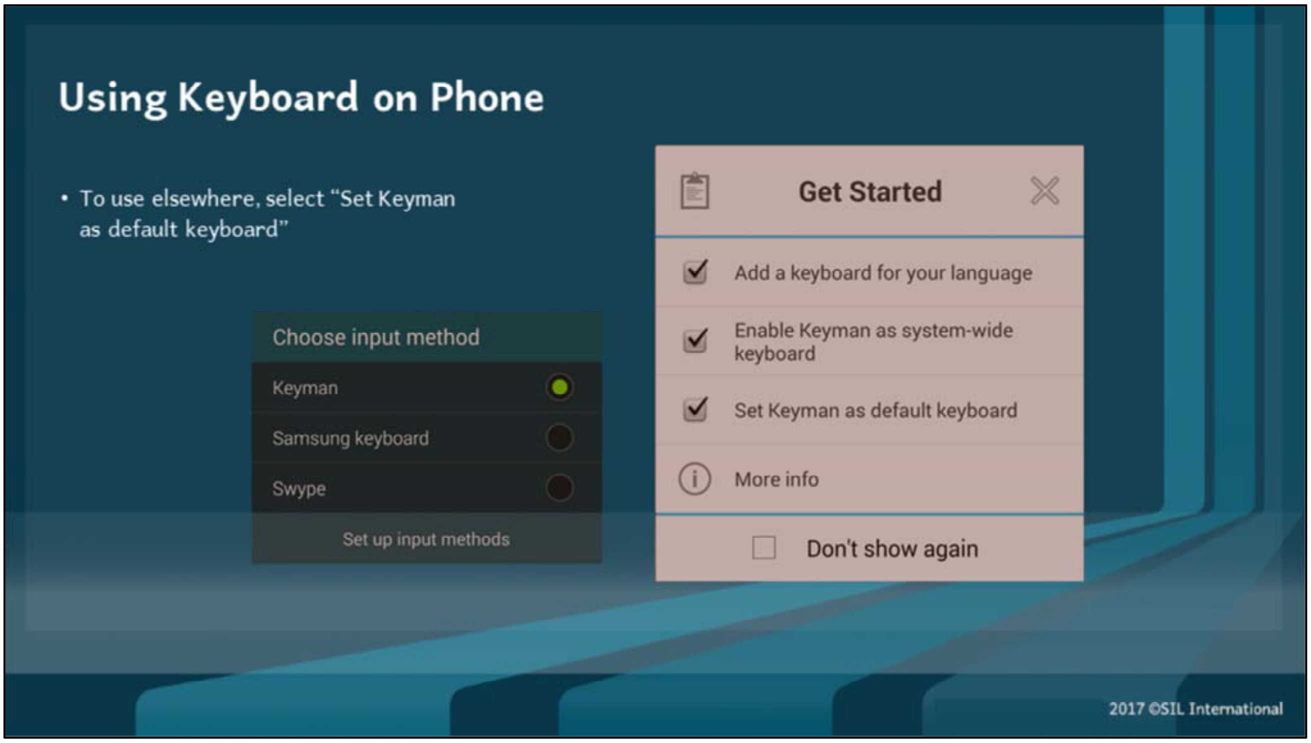


Using Keyboard on Phone

- Click on “world” icon

2017 ©SIL International

Since we’ve opened the Keyman app and selected our keyboard, we can now type in this application. Then you will need to copy and paste into wherever you want the text. This may not be what you want if you wish to use the keyboard in many applications on the phone.



Using Keyboard on Phone

- To use elsewhere, select "Set Keyman as default keyboard"

Choose input method

Keyman ●

Samsung keyboard ○

Swype ○

Set up input methods

Get Started

☑ Add a keyboard for your language

☑ Enable Keyman as system-wide keyboard

☑ Set Keyman as default keyboard

More info

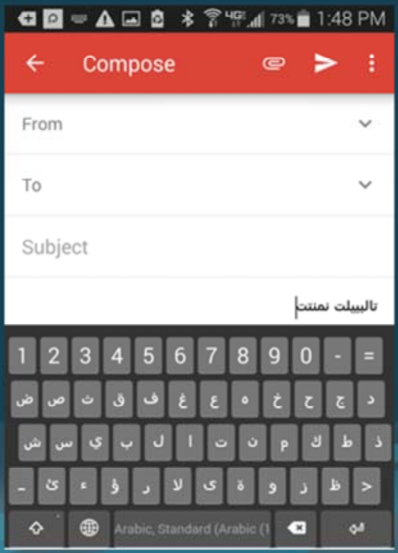
□ Don't show again

2017 ©SIL International

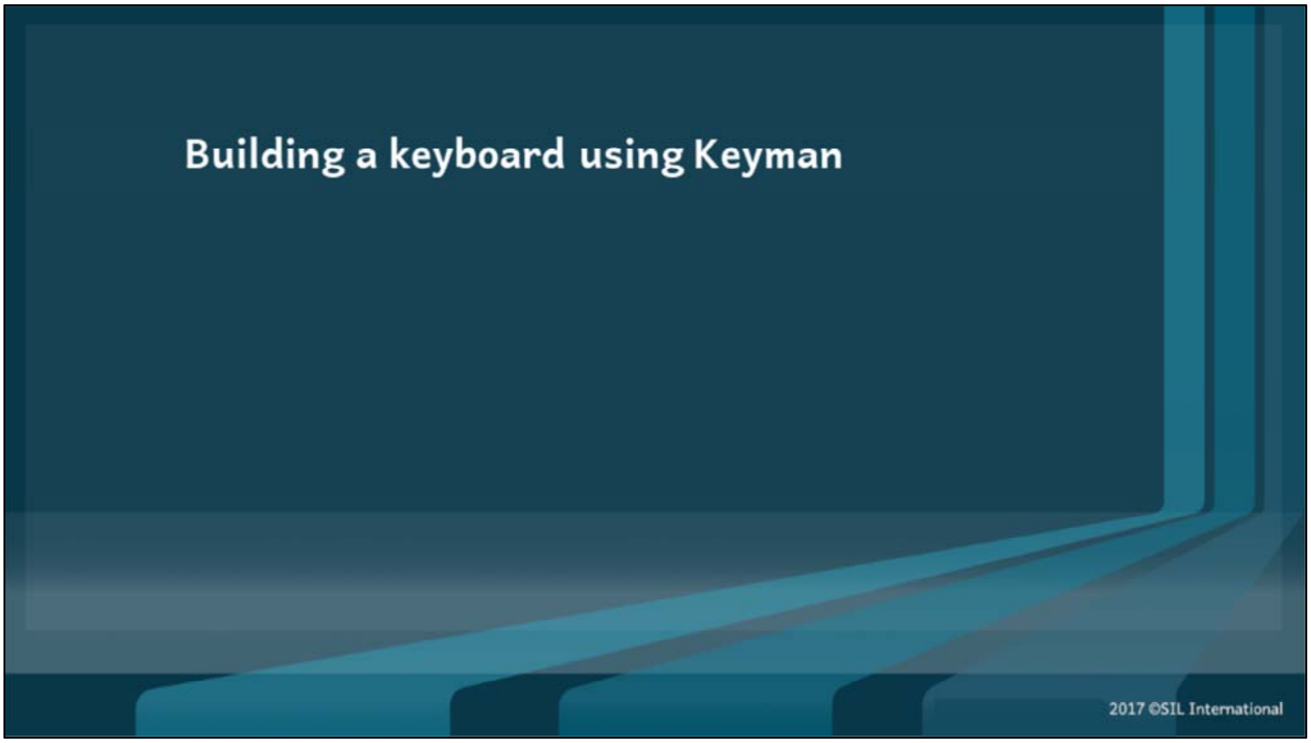
In that case, you will want to set Keyman as your default keyboard.
Then you will need to "Choose input method" and you should select "Keyman"

Using Keyboard on Phone

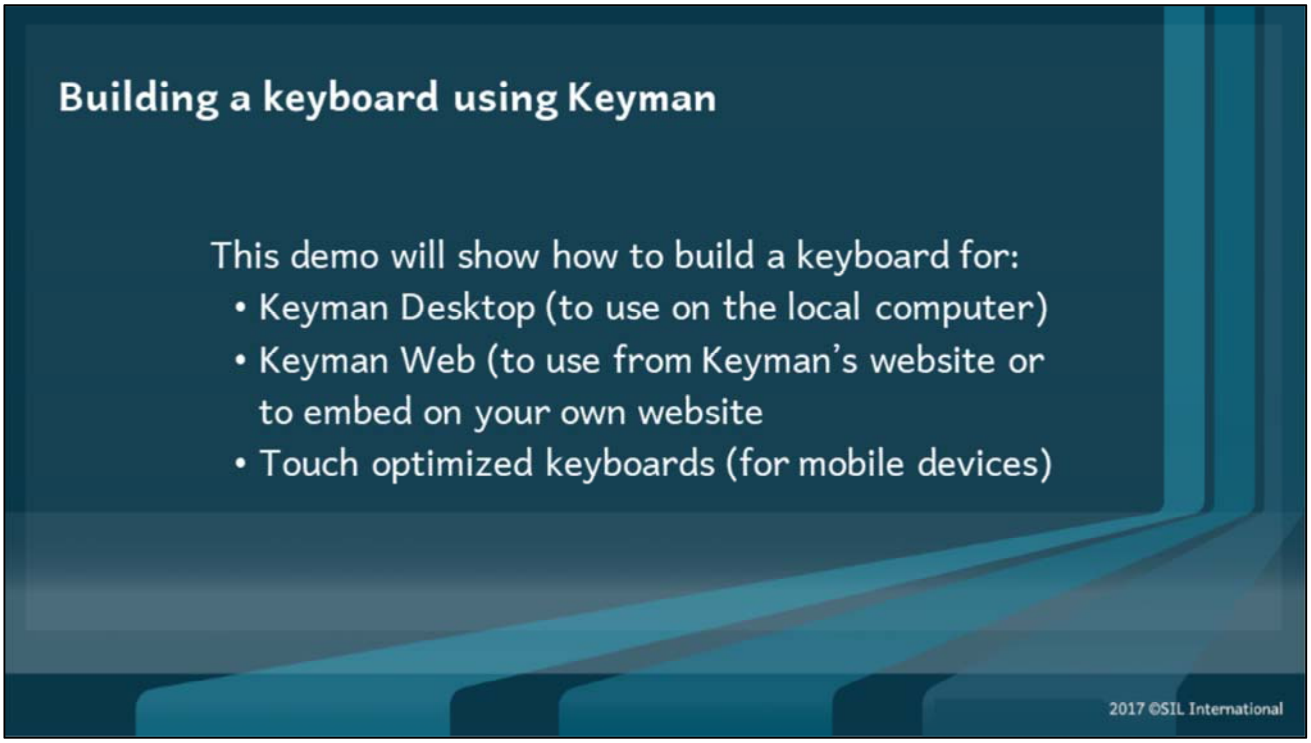
- Keyman is now available in other applications
  - Click on the world button to set your keyboard



تالبيلت نمنتت



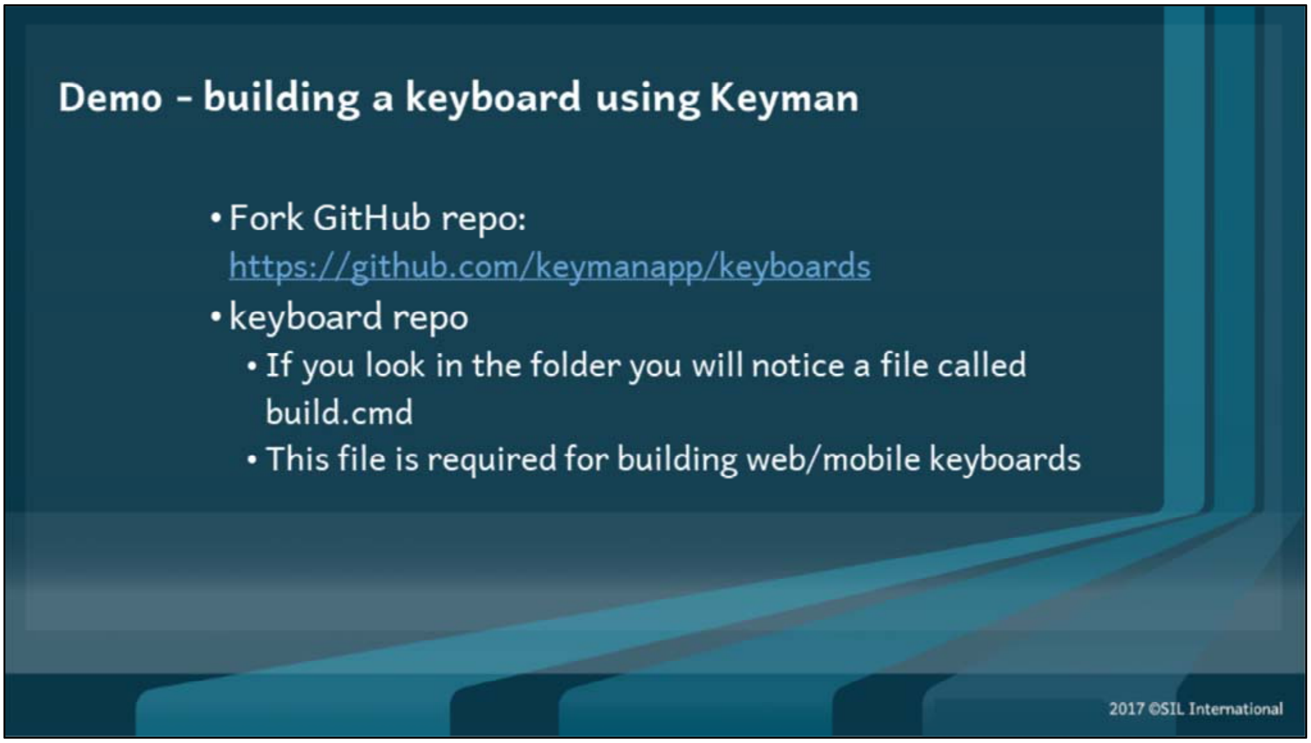
Building a keyboard using Keyman



Building a keyboard using Keyman

This demo will show how to build a keyboard for:

- Keyman Desktop (to use on the local computer)
- Keyman Web (to use from Keyman's website or to embed on your own website
- Touch optimized keyboards (for mobile devices)



Demo - building a keyboard using Keyman

- Fork GitHub repo: https://github.com/keymanapp/keyboards
- keyboard repo
  - If you look in the folder you will notice a file called build.cmd
  - This file is required for building web/mobile keyboards

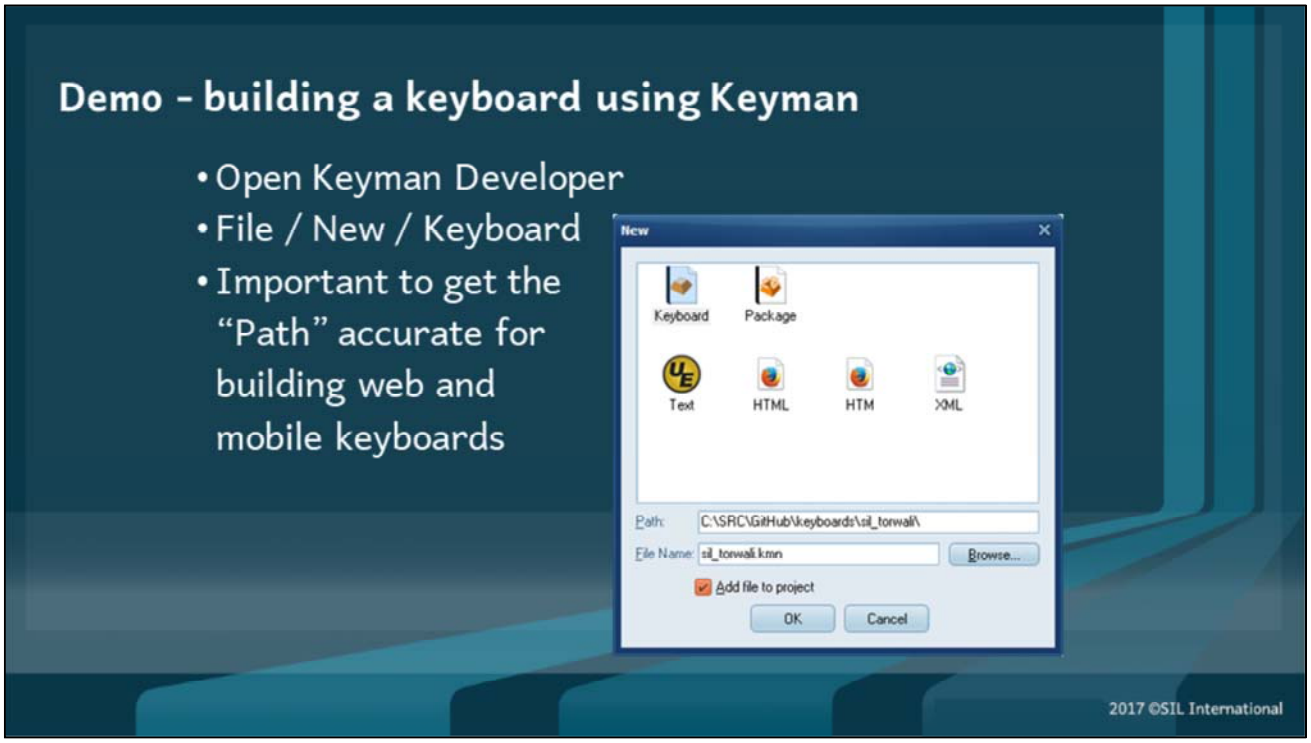
2017 ©SIL International

If all you want to do is build a keyboard and distribute the Keyman package (.kmp) through your own website or through Keyman.com, you can just create, compile and package it up on your computer.

However, if you want to put your keyboard up on KeymanWeb or Android you will need to go through the process of getting set up on GitHub, forking the keyboards repo on GitHub and submitting your keyboard files through GitHub.

Some of this will seem pretty technical. However, it is important to do this if you want to redistribute your keyboard or develop a keyboard for web or mobile use.

This session will not discuss how to fork a repo. However, we can help you do that during the workshop session.



Demo - building a keyboard using Keyman

- Open Keyman Developer
- File / New / Keyboard
- Important to get the "Path" accurate for building web and mobile keyboards

Path: C:\SRC\GitHub\keyboards\sil_torwali\

File Name: sil_torwali.kmn

☑ Add file to project

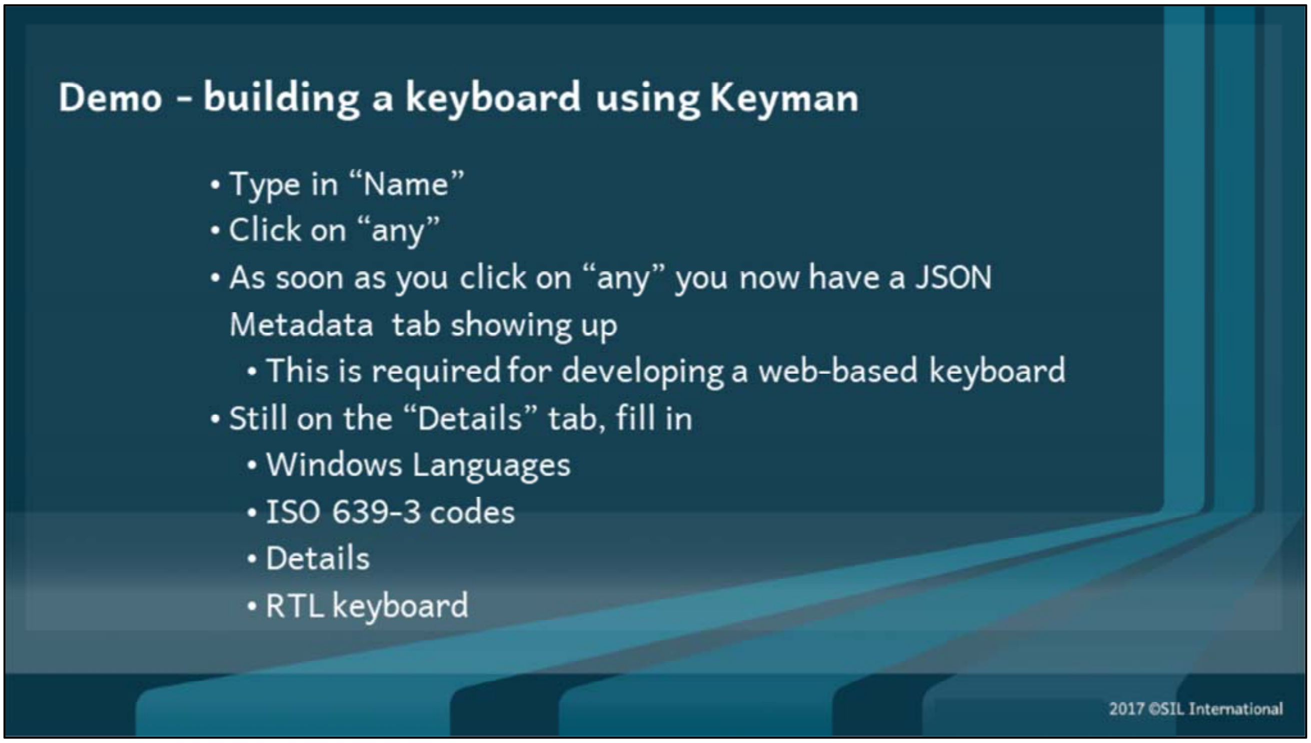
2017 ©SIL International

The path here is where you created a fork (and clone) of the keyboards repo.

This path is used for where Keyman creates all the files it needs

Folder and Keyboard name should match

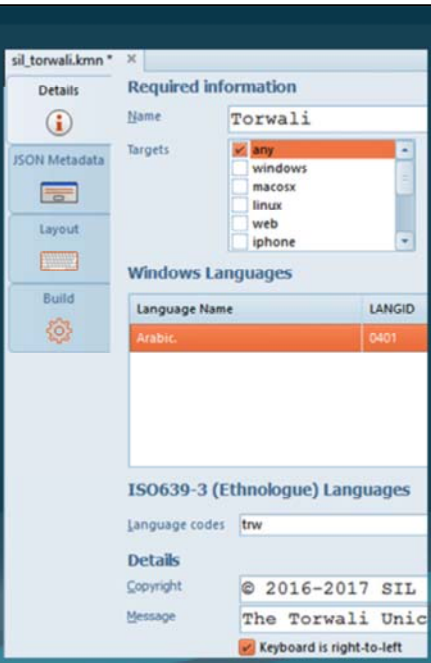
Only special character allowed is "_"



Demo - building a keyboard using Keyman

- Type in “Name”
- Click on “any”
- As soon as you click on “any” you now have a JSON Metadata tab showing up
  - This is required for developing a web-based keyboard
- Still on the “Details” tab, fill in
  - Windows Languages
  - ISO 639-3 codes
  - Details
  - RTL keyboard

Demo



Required information

Name: Torwali

Targets:

- ☑ any
- □ windows
- □ macosx
- □ linux
- □ web
- □ iphone

Windows Languages

Language Name	LANGID
Arabic.	0401

ISO639-3 (Ethnologue) Languages

Language codes: trw

Details

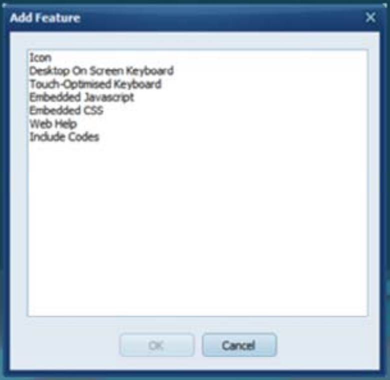
Copyright: © 2016-2017 SIL

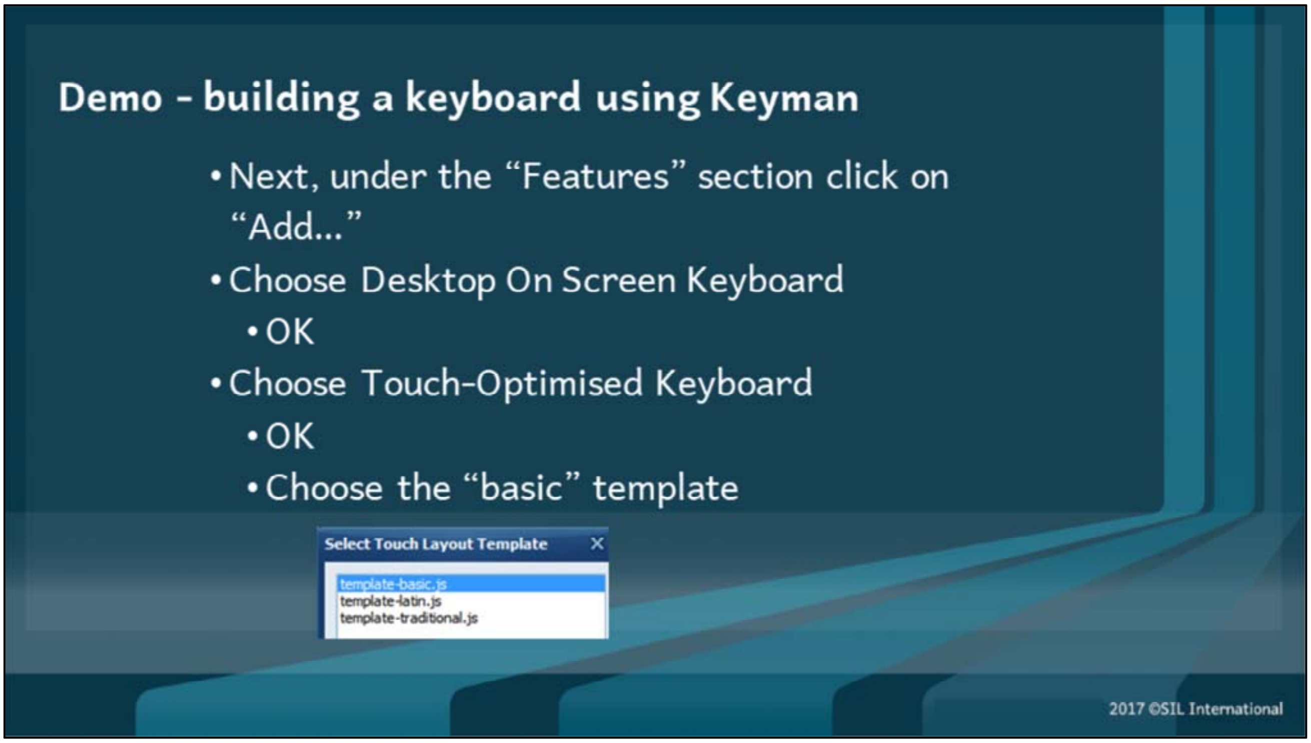
Message: The Torwali Unic

☑ Keyboard is right-to-left

Demo - building a keyboard using Keyman

- Next, under the “Features” section click on “Add...”
- Choose Desktop On Screen Keyboard
  - OK
- Choose Touch-Optimised Keyboard
  - OK





Demo - building a keyboard using Keyman

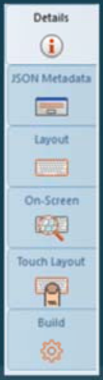
- Next, under the “Features” section click on “Add…”
- Choose Desktop On Screen Keyboard
  - OK
- Choose Touch-Optimised Keyboard
  - OK
  - Choose the “basic” template

Select Touch Layout Template

template-basic.js
template-latin.js
template-traditional.js

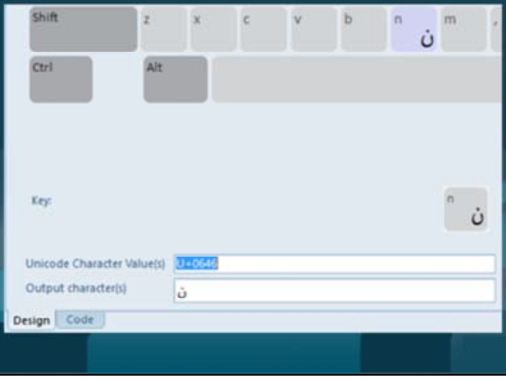
Demo - building a keyboard using Keyman

- Adding those has now added an "On-screen" tab and a "Touch Layout" tab
- You can also add an "Icon" if you wish from the Features menu



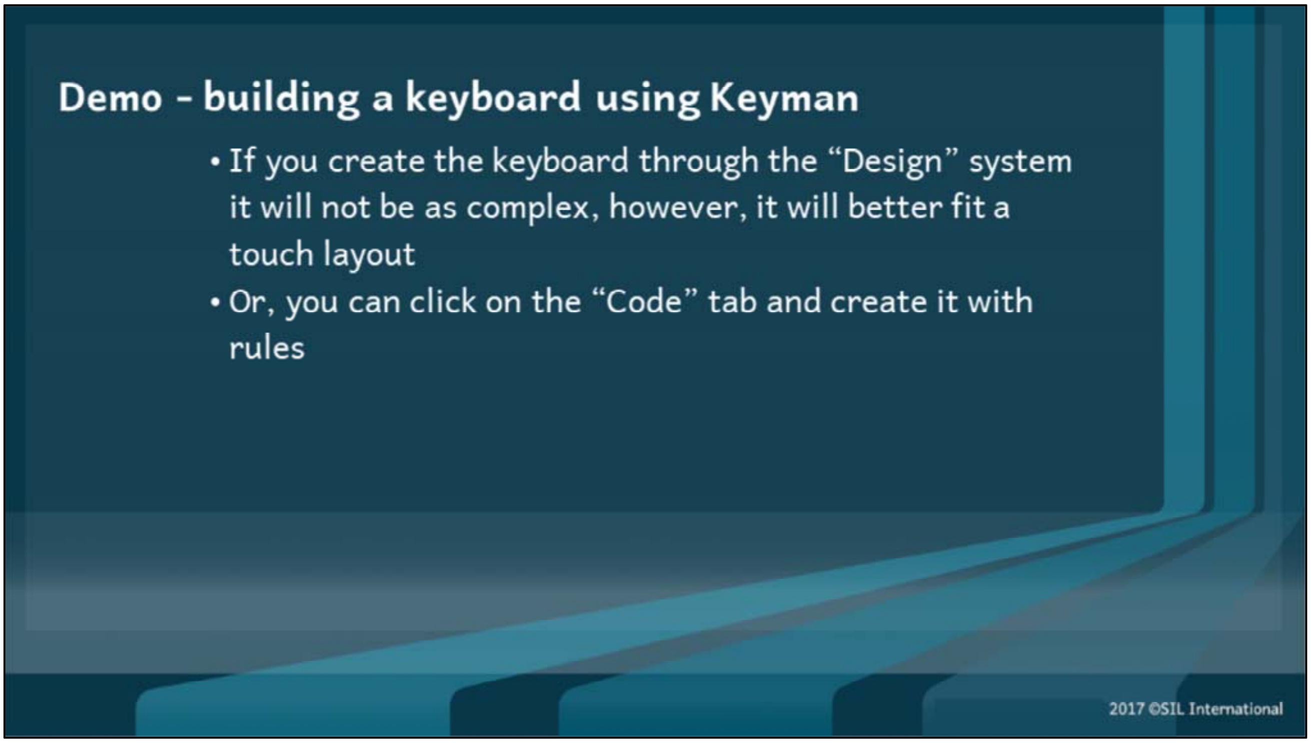
Demo – building a keyboard using Keyman

- Click on the “Layout” tab
- You can either create a keyboard visually using the “Design” tab
  - Click on the key and then type in the Unicode Character Value



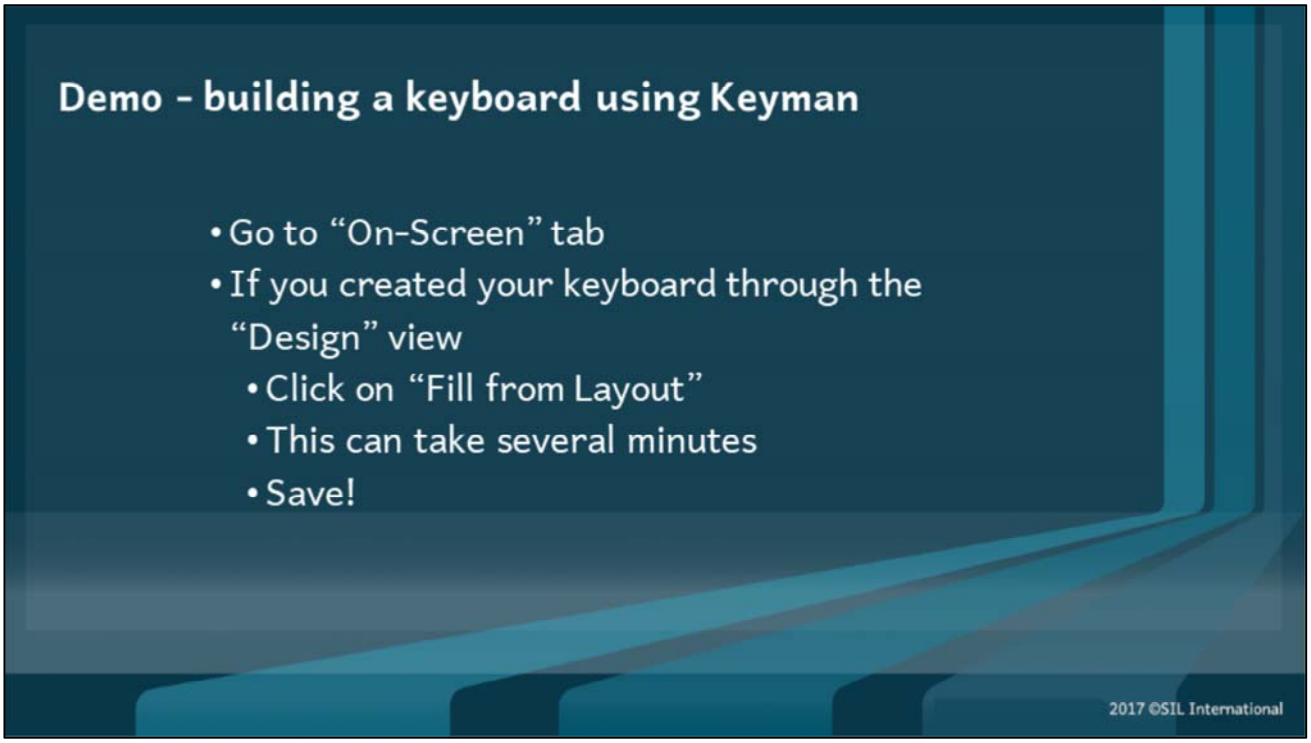
Unicode Character Value(s) U+0646

Output character(s) ن



Demo - building a keyboard using Keyman

- If you create the keyboard through the “Design” system it will not be as complex, however, it will better fit a touch layout
- Or, you can click on the “Code” tab and create it with rules



Demo - building a keyboard using Keyman

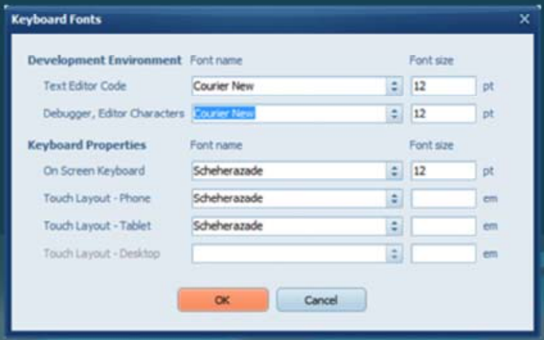
- Go to “On-Screen” tab
- If you created your keyboard through the “Design” view
  - Click on “Fill from Layout”
  - This can take several minutes
  - Save!

2017 ©SIL International

Sometimes “Fill from Layout” takes a long time. I’ve clicked on the “x” and then clicked on “Fill from Layout” again and it immediately builds.

Demo - building a keyboard using Keyman

- Go to “Keyboard / Fonts”
- Select the font you want displayed on the Keyboard layout

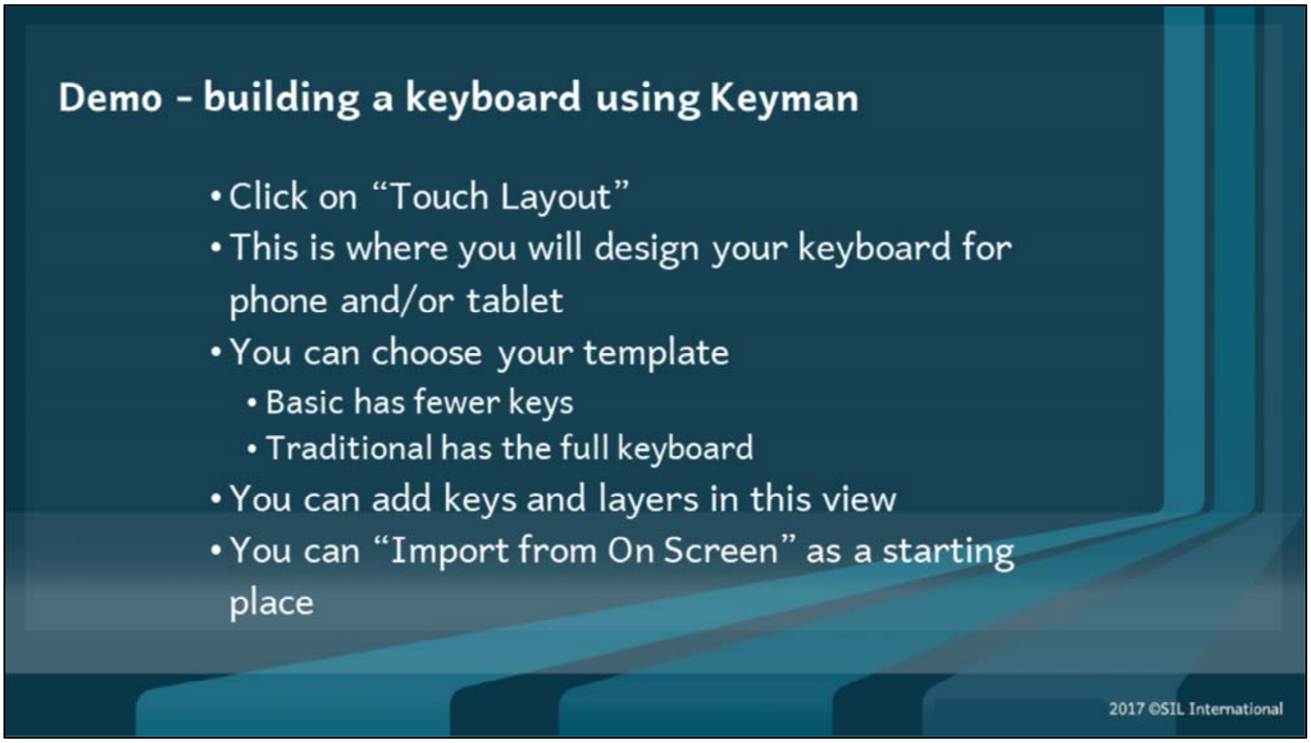


Keyboard Fonts

Development Environment	Font name	Font size	
Text Editor Code	Courier New	12	pt
Debugger, Editor Characters	Courier New	12	pt
Keyboard Properties	Font name	Font size	
On Screen Keyboard	Scheherazade	12	pt
Touch Layout - Phone	Scheherazade		em
Touch Layout - Tablet	Scheherazade		em
Touch Layout - Desktop			em

2017 ©SIL International

Sometimes “Fill from Layout” takes a long time. I’ve clicked on the “x” and then clicked on “Fill from Layout” again and it immediately builds.



Demo - building a keyboard using Keyman

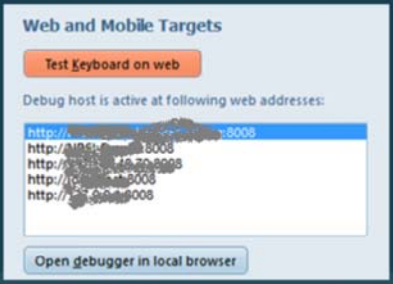
- Click on “Touch Layout”
- This is where you will design your keyboard for phone and/or tablet
- You can choose your template
  - Basic has fewer keys
  - Traditional has the full keyboard
- You can add keys and layers in this view
- You can “Import from On Screen” as a starting place

2017 ©SIL International

Basic has fewer keys. It is easier to use on a phone
Traditional has the full keyboard. It might work on a tablet, but it would not be useable on a phone

Demo – building a keyboard using Keyman

- Once you've designed all the keyboard layouts
  - Save
  - Go to "Build"
  - Compile your keyboard
  - Test your keyboard for the Desktop
  - Test Keyboard on web
    - Click on url
    - Click on "Open debugger in local browser"



Debug host is active at following web addresses:

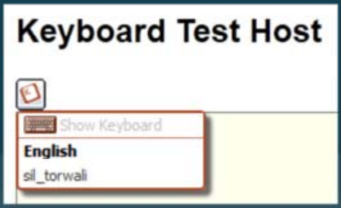
```
http://[redacted]:8008
http://[redacted]8008
http://[redacted]8008
http://[redacted]8008
http://[redacted]8008
```

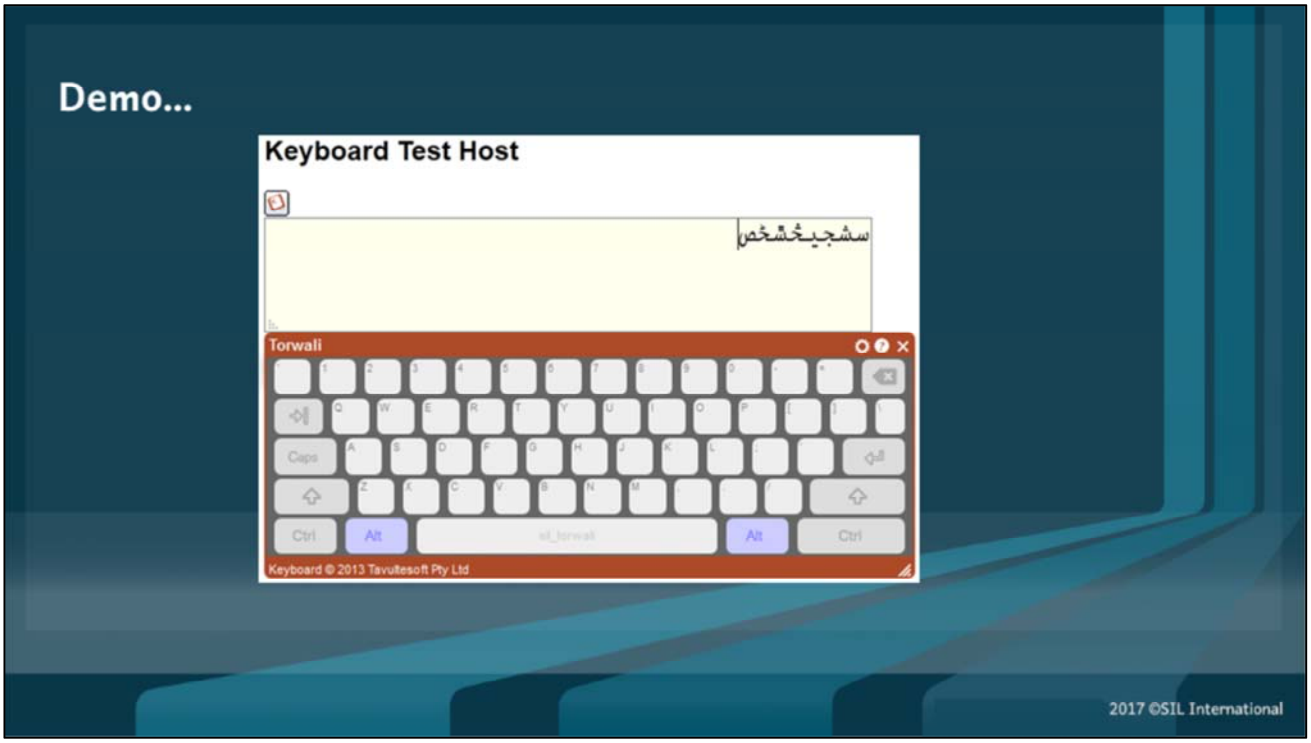
When compiling: If you differentiated between RALT and LALT in your Desktop keyboard you will get some warning messages for KeymanWeb.

When you click on "Test Keyboard on web" it generates some URLs for testing purposes. Click on a URL

Demo - building a keyboard using Keyman

- In order to test, click on the little Keyman icon and choose your keyboard





Demo...

Keyboard Test Host

سشجیـځشخص

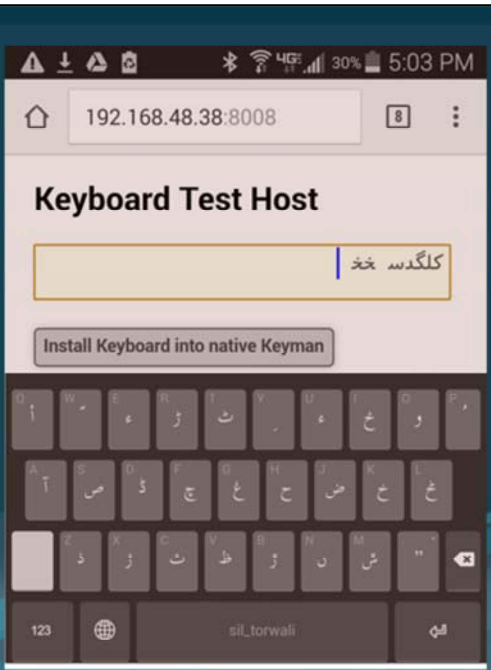
Keyboard © 2013 Tavultesoft Pty Ltd

2017 ©SIL International

This is testing your keyboard for KeymanWeb.

Demo

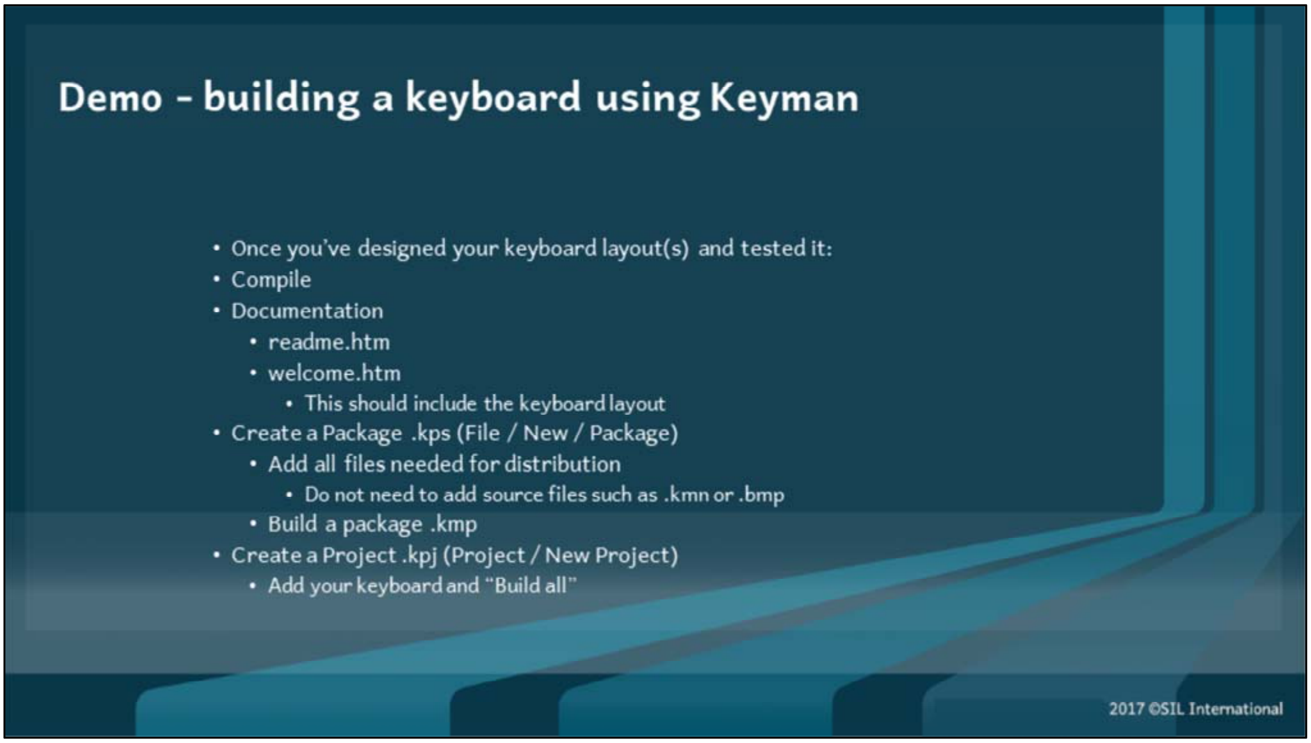
- To test on your phone
  - *Must be on the same network as your computer*
- On your phone, type in the URL (http://192.168.48...:8008)



192.168.48.38:8008

Keyboard Test Host

کلگدس خخ

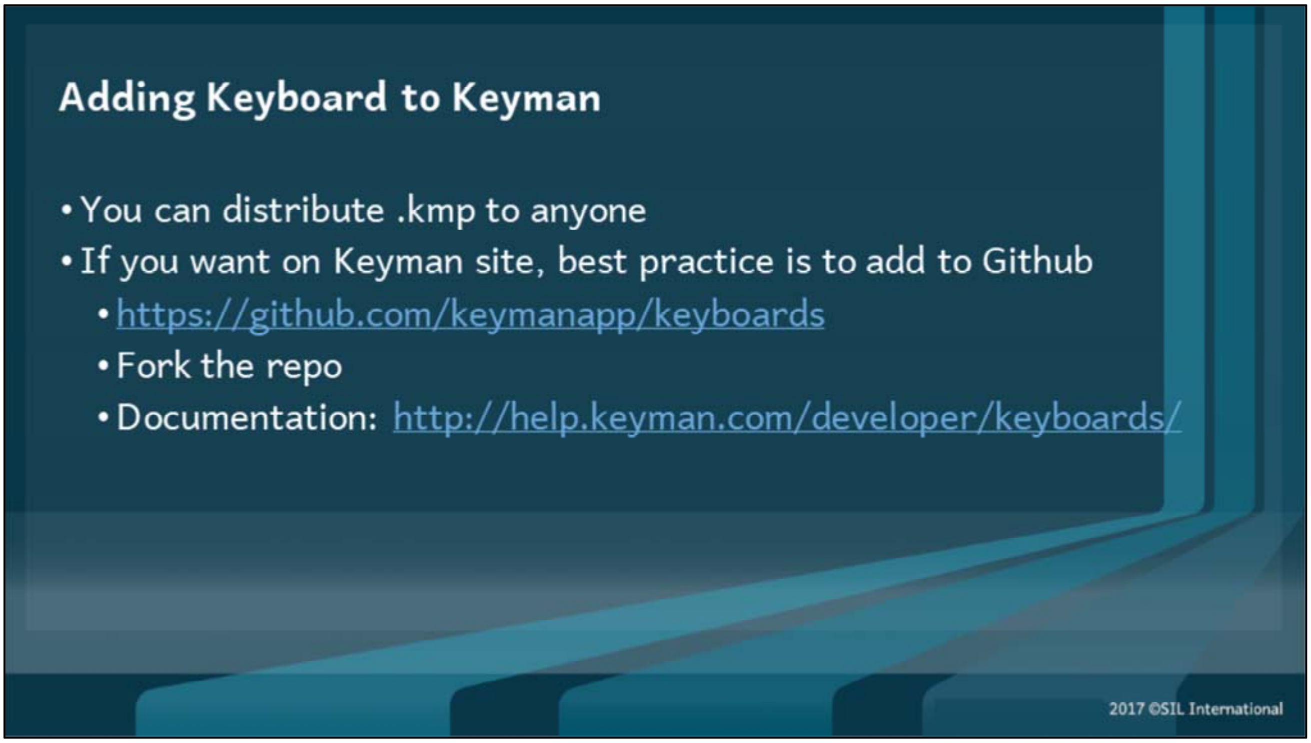


Demo - building a keyboard using Keyman

- Once you've designed your keyboard layout(s) and tested it:
- Compile
- Documentation
  - readme.htm
  - welcome.htm
    - This should include the keyboard layout
- Create a Package .kps (File / New / Package)
  - Add all files needed for distribution
    - Do not need to add source files such as .kmn or .bmp
  - Build a package .kmp
- Create a Project .kpj (Project / New Project)
  - Add your keyboard and "Build all"

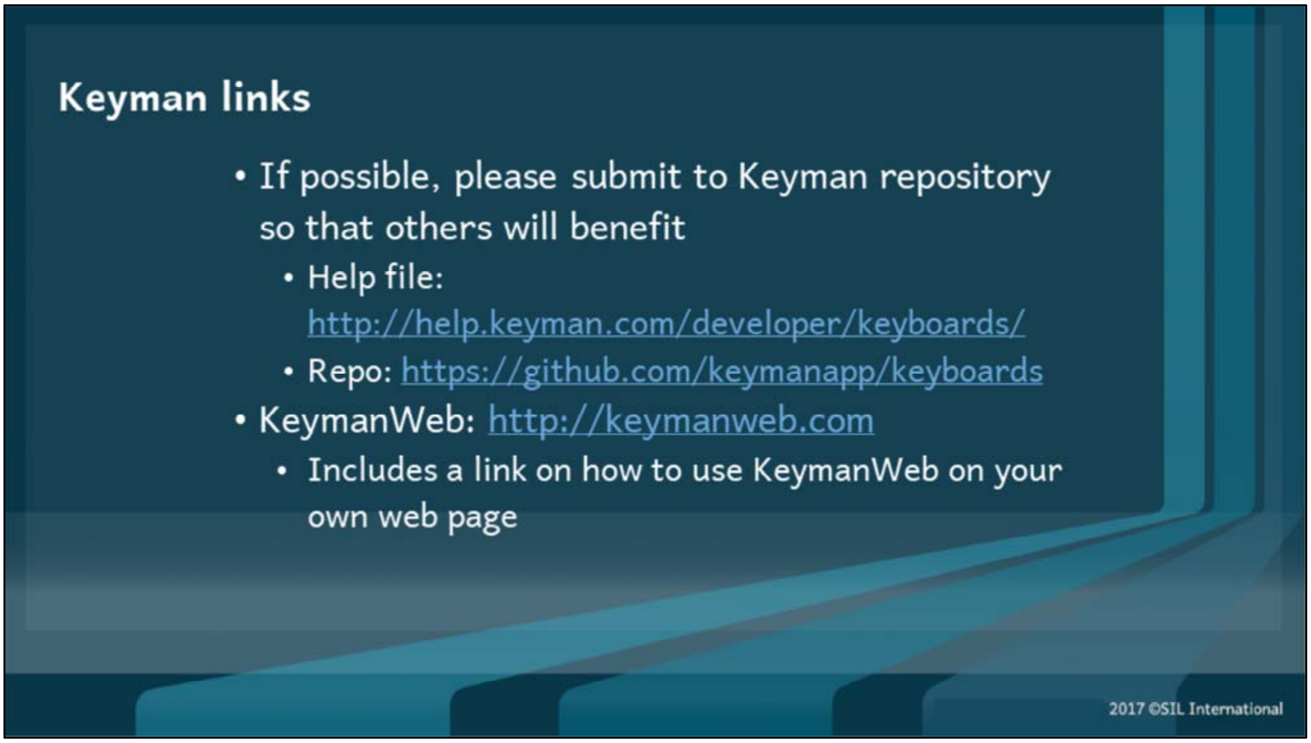
All of these are required for adding to Github

Adding Keyboard to Keyman



Adding Keyboard to Keyman

- You can distribute .kmp to anyone
- If you want on Keyman site, best practice is to add to Github
  - https://github.com/keymanapp/keyboards
  - Fork the repo
  - Documentation: http://help.keyman.com/developer/keyboards/

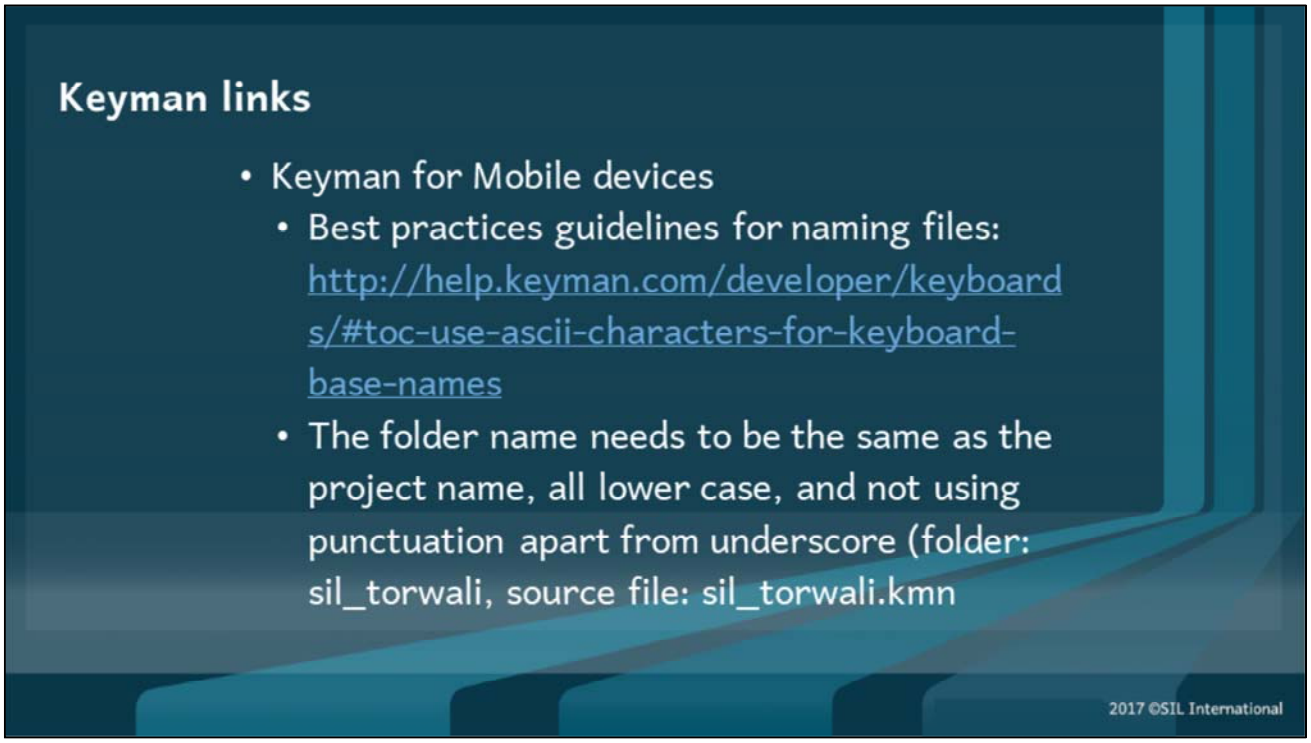


Keyman links

- If possible, please submit to Keyman repository so that others will benefit
  - Help file: http://help.keyman.com/developer/keyboards/
  - Repo: https://github.com/keymanapp/keyboards
- KeymanWeb: http://keymanweb.com
  - Includes a link on how to use KeymanWeb on your own web page

2017 ©SIL International

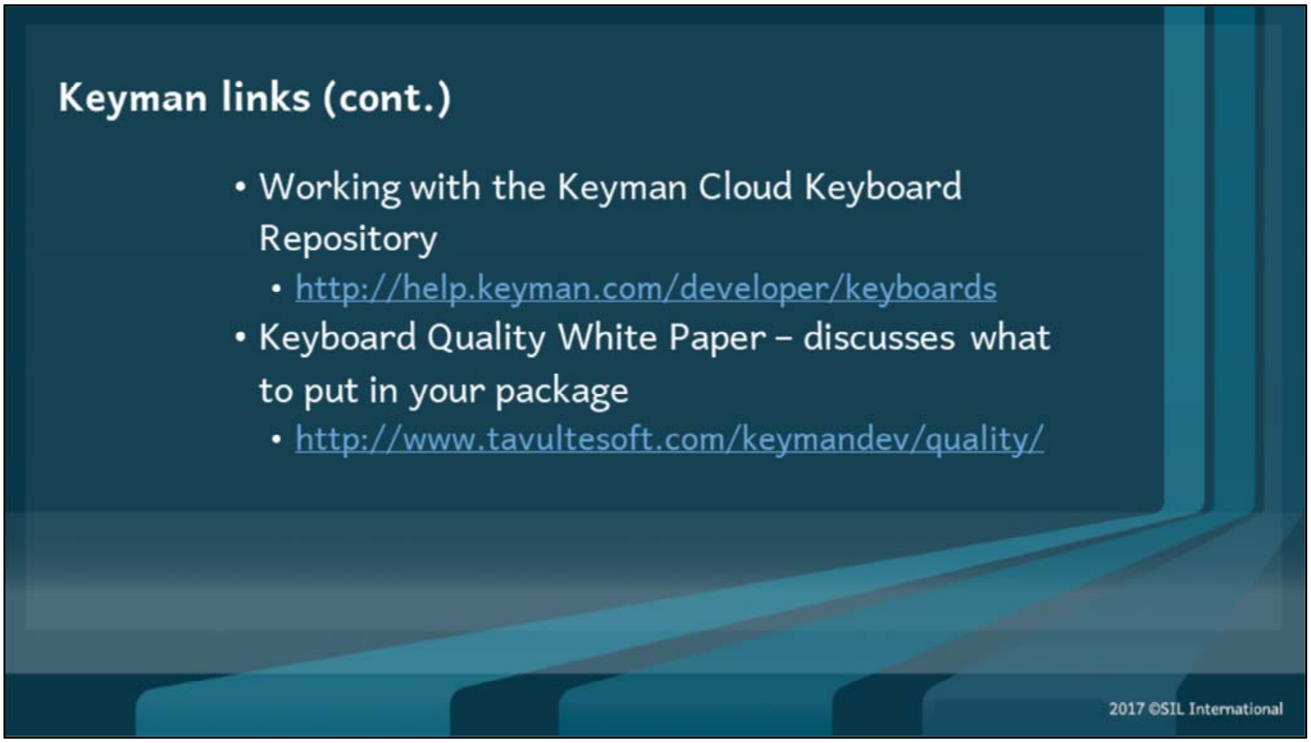
The folder name needs to be the same as the project name, all lower case, and not using punctuation apart from underscore (this is so the name can be a valid identifier in Javascript and we don't run into casing issues across operating systems). I suggest renaming the .kmn, .kpj, .kps and folder to **sil_torwali**) You'll need to update the file references in the .kpj and .kps files. It's a good idea to rename the other related files as well for consistency -- sil_nubian.bmp; sil_nubian-layout.js, sil_nubian-1.0.json.



Keyman links

- Keyman for Mobile devices
  - Best practices guidelines for naming files: http://help.keyman.com/developer/keyboards/#toc-use-ascii-characters-for-keyboard-base-names
  - The folder name needs to be the same as the project name, all lower case, and not using punctuation apart from underscore (folder: sil_torwali, source file: sil_torwali.kmn

The folder name needs to be the same as the project name, all lower case, and not using punctuation apart from underscore (this is so the name can be a valid identifier in Javascript and we don't run into casing issues across operating systems). I suggest renaming the .kmn, .kpj, .kps and folder to **sil_torwali**) You'll need to update the file references in the .kpj and .kps files. It's a good idea to rename the other related files as well for consistency -- sil_torwali.bmp; sil_torwali-layout.js, sil_torwali-1.0.json.

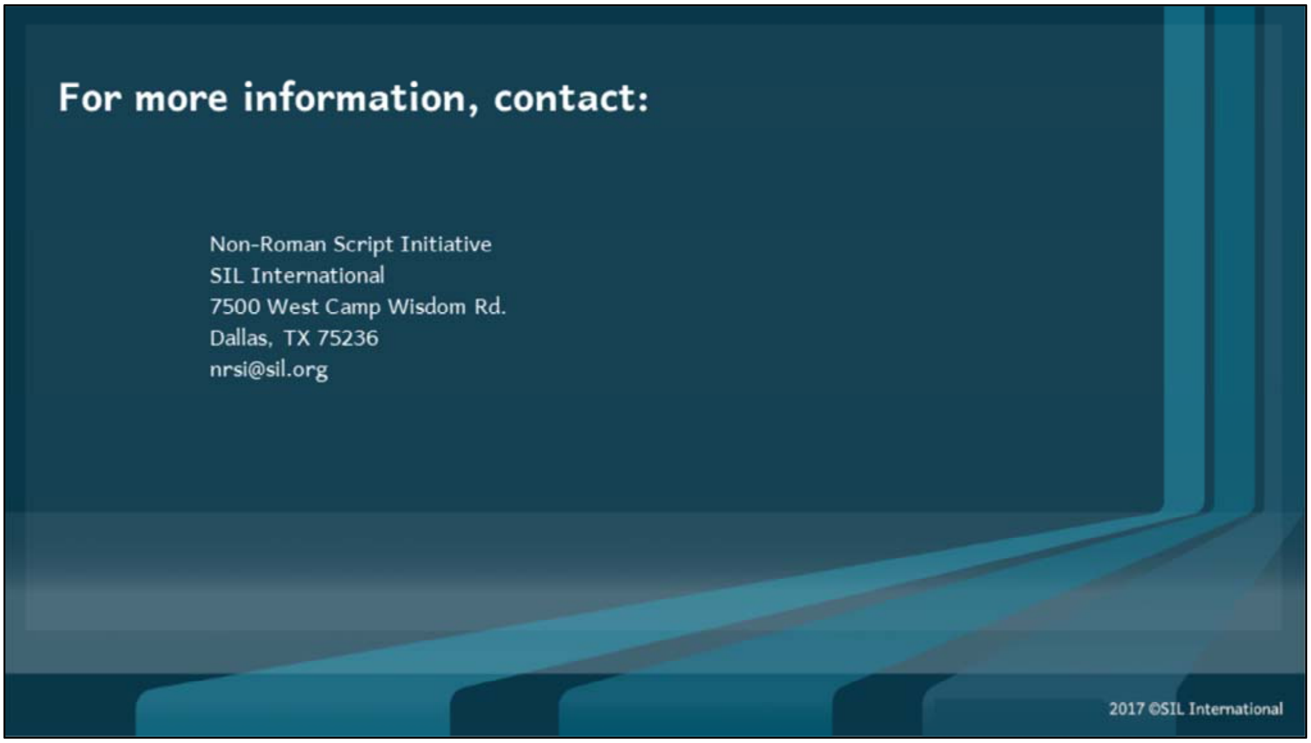


Keyman links (cont.)

- Working with the Keyman Cloud Keyboard Repository
  - http://help.keyman.com/developer/keyboards
- Keyboard Quality White Paper – discusses what to put in your package
  - http://www.tavultesoft.com/keymandev/quality/

2017 ©SIL International

Consider whether to include fonts. If most computers have the fonts you need, there is no need to include the fonts in the keyboard package. However, if very few fonts contain the characters you need you might consider including the fonts in the package (as long as the license allow for that). The downside is that when fonts are updated you will need to remember to update the keyboard package as well.



For more information, contact:

Non-Roman Script Initiative
SIL International
7500 West Camp Wisdom Rd.
Dallas, TX 75236
nrsi@sil.org