Rethinking Font development workflow

Aravinda VK
http://aravindavk.in
@aravindavk

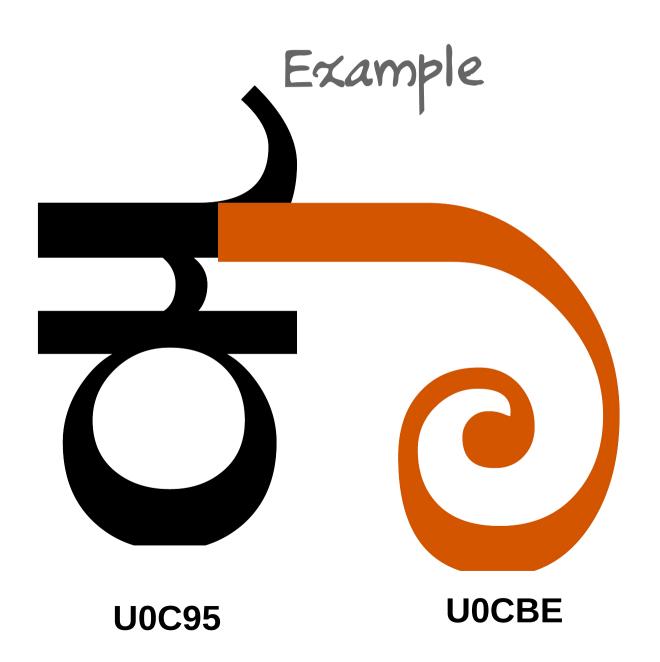# Multiple fonts and common rules

- Glyph names should be same accross the fonts

- GSUB remains same accross the fonts, only GPOS changes
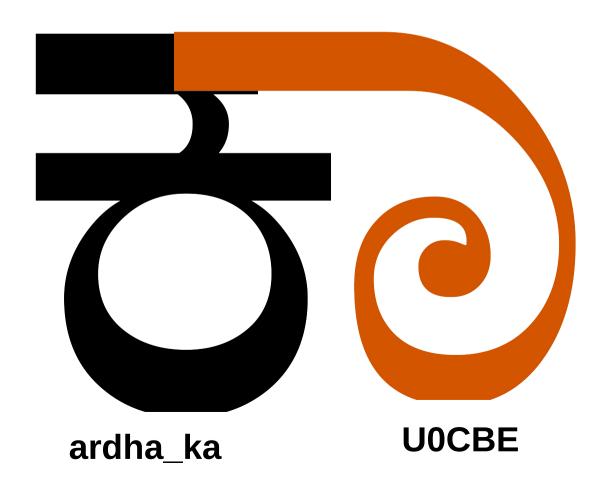
Fonts with bad glyph names!

- Cannot be merged with other fonts

- Automation is very difficult

Adobe Featurefile

http://www.adobe.com/devnet/opentype/afdko/topic_feature_file_syntax.html

Example

U0C95   U0CBE

Example

ardha_ka U0CBE

With Featurefile

```
lookup abvs_kan {
    sub \U0C95' \U0CBE by \ardha_ka;
} abvs_kan;


feature abvs {
  script knd2;
    language dflt ;
    lookup abvs_kan;

  script knda;
    language dflt ;
    lookup abvs_kan;
} abvs;
```

Get existing Featurefile

```python
#!/usr/bin/python
import fontforge

font = fontforge.open("lohit-kannada.sfd")
font.generateFeatureFile("lohit-kannada.fea")
```

Apply modified Featurefile

```python
#!/usr/bin/python
import fontforge

font = fontforge.open("lohit-kannada.sfd")
# Remove all GSUB lookups
for lookup in font.gsub_lookups:
    font.removeLookup(lookup)
# Remove all GPOS lookups
for lookup in font.gpos_lookups:
    font.removeLookup(lookup)
# Merge the new featurefile
font.mergeFeature("lohit-kannada.fea")
font.save()
```

Questions?

More Python Fontforge scripts available in
https://github.com/aravindavk/fontforge-python-cookbook

https://github.com/aravindavk/Gubbi

Aravinda VK
http://aravindavk.in
@aravindavk