مسائل الخطوط العربية

Arabic Fonts Issues

Mamoun Sakkal

VOLT DAY

@ Microsoft

June 2, 2005

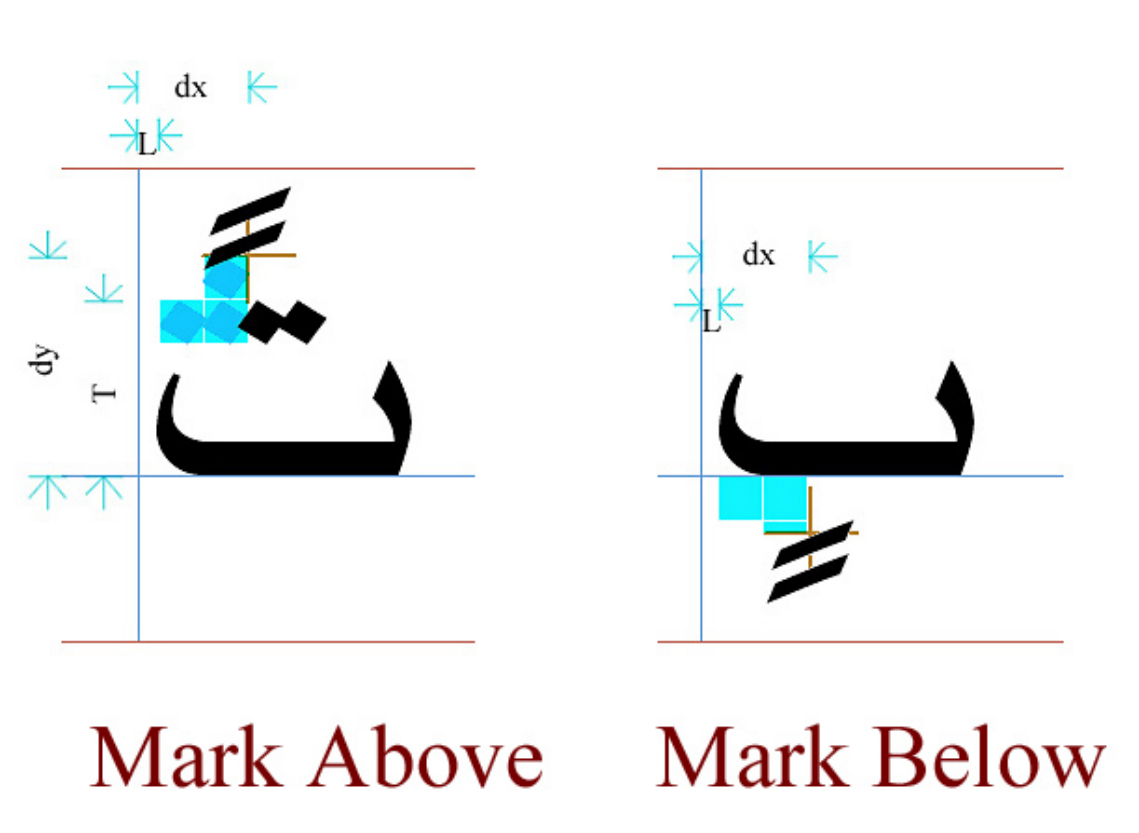
Updated January 31, 2015

Arabic Fonts Issues

- Marks Positioning.
- Use of Groups.
- Components Instead of Ligatures.
- Urdu Ye Barree Kerning.
- VOLT, OpenType and Arabic.

Marks Positioning

- We define the position of marks in relation to the glyph outlines using dot units.



Marks Positioning

- We create a lookup file with the values of all the anchor points for the glyphs in the font based on rules we develop for each font style.
- We then import the lookup to VOLT to position the marks on each glyph.
- This insures the positioning of the marks on similar letters in the exact same positions.
- This also allows us to revise the rules and re-use them from font to font without starting new every time.

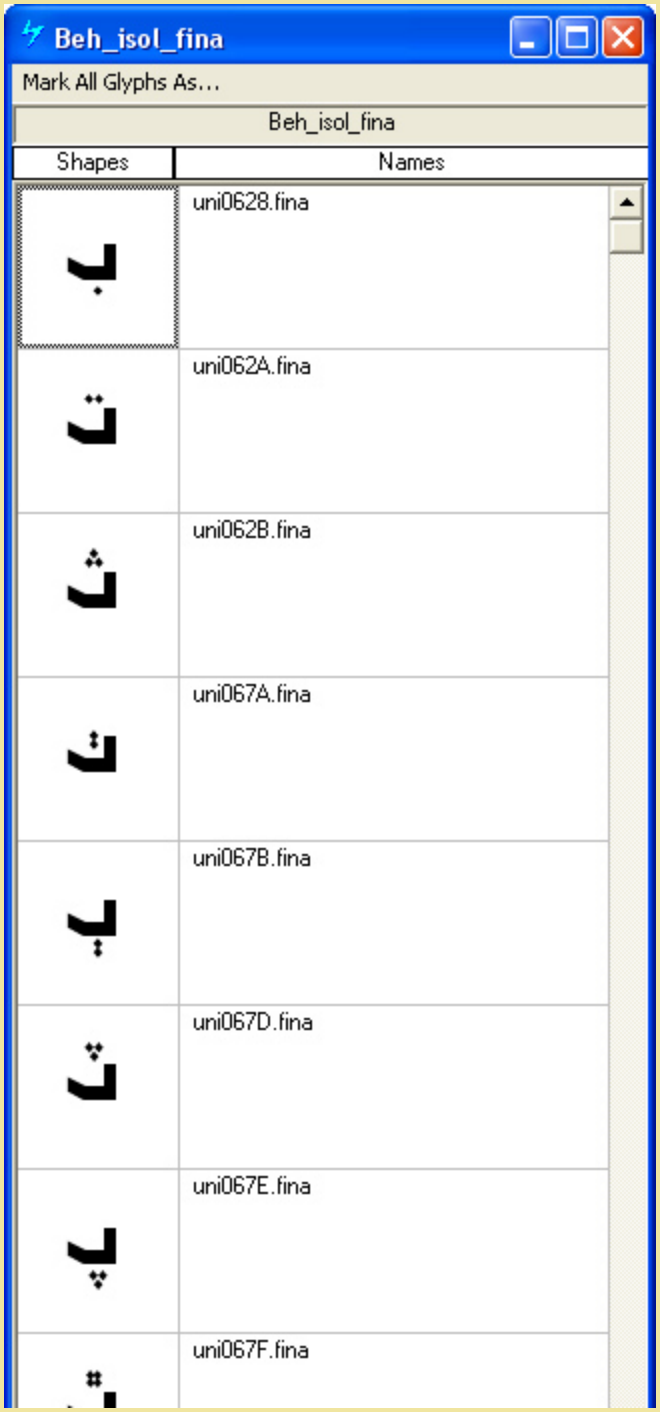
Marks Positioning

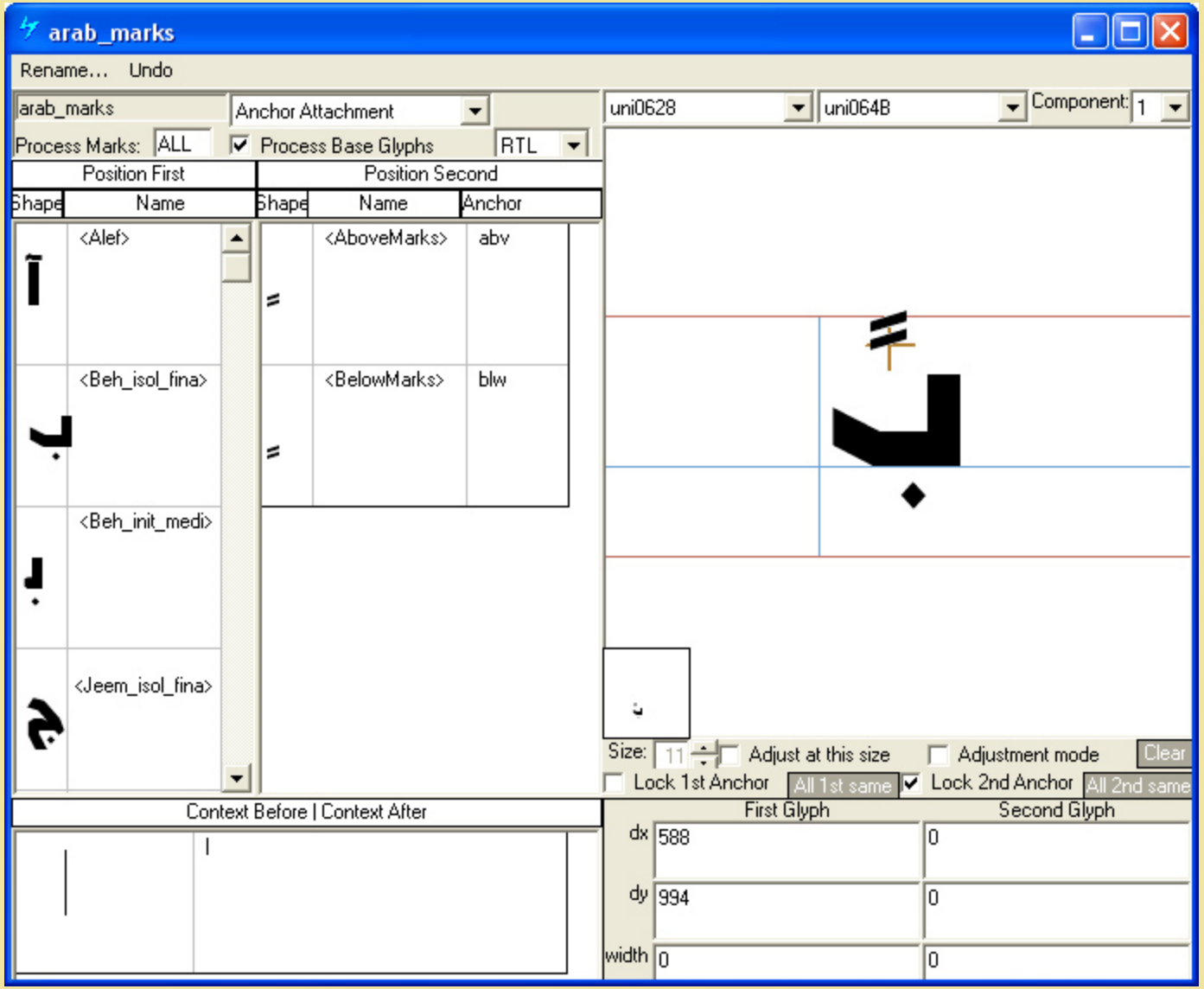
- The uniform treatment of marks positioning over base glyphs is important in Arabic fonts because vowel marks are necessary in every day writing.
- In "**Arabic Typesetting**" font, when updated to cover Unicode 4.1 Arabic block glyphs, we will have the following mark/glyph combinations:
- 868 base glyphs x 58 marks = 50,344 positions
- 1,068 ligature glyphs x 2 components x 58 marks = 123,888 positions
- **Total positions 174,232**: Too many to do manually.

Groups

- We use groups to make sure we cover all the glyphs that require mark positioning.
- Similar glyphs in a group will make mark positioning easier when done manually.

Sakkal Baseet Heavy

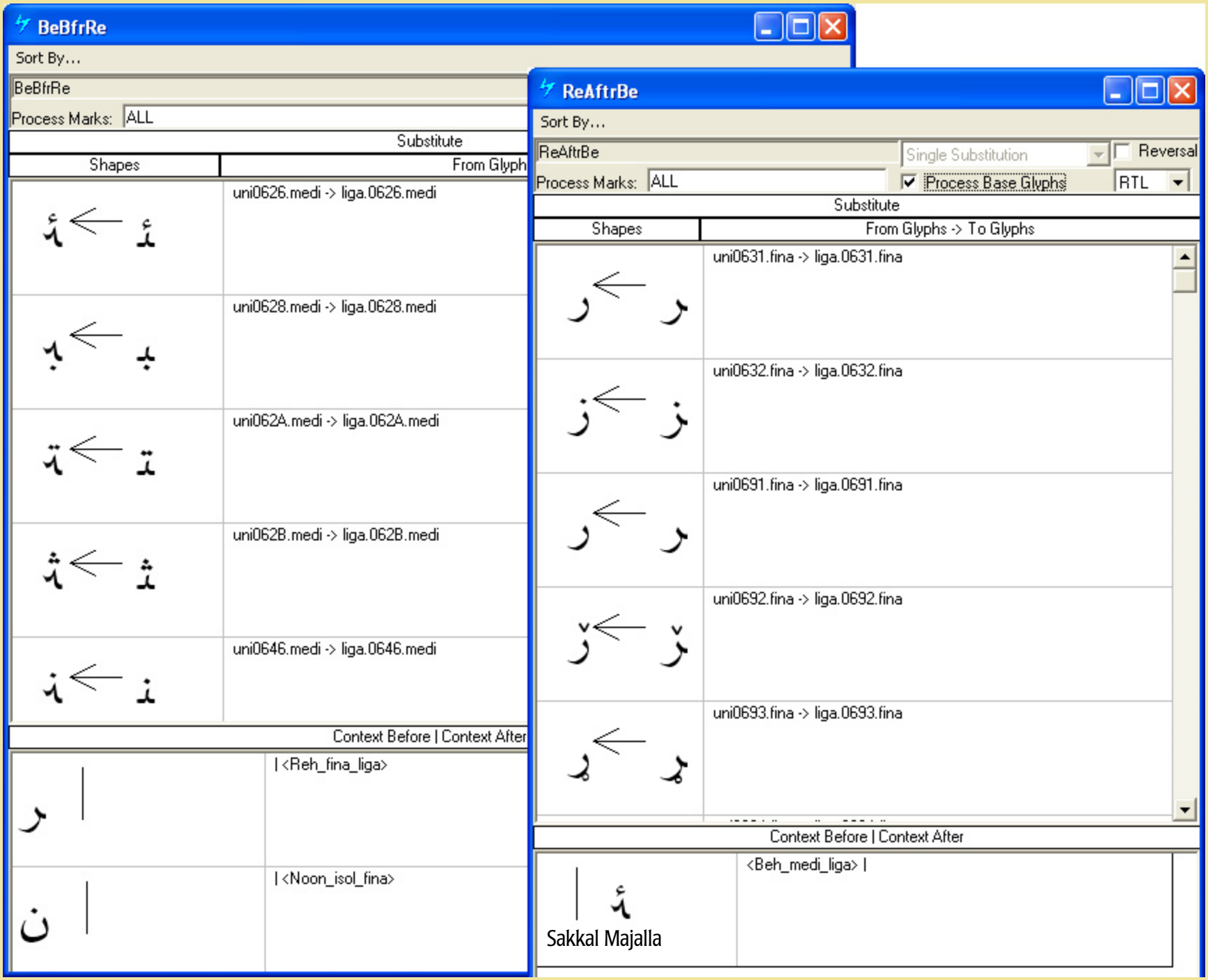


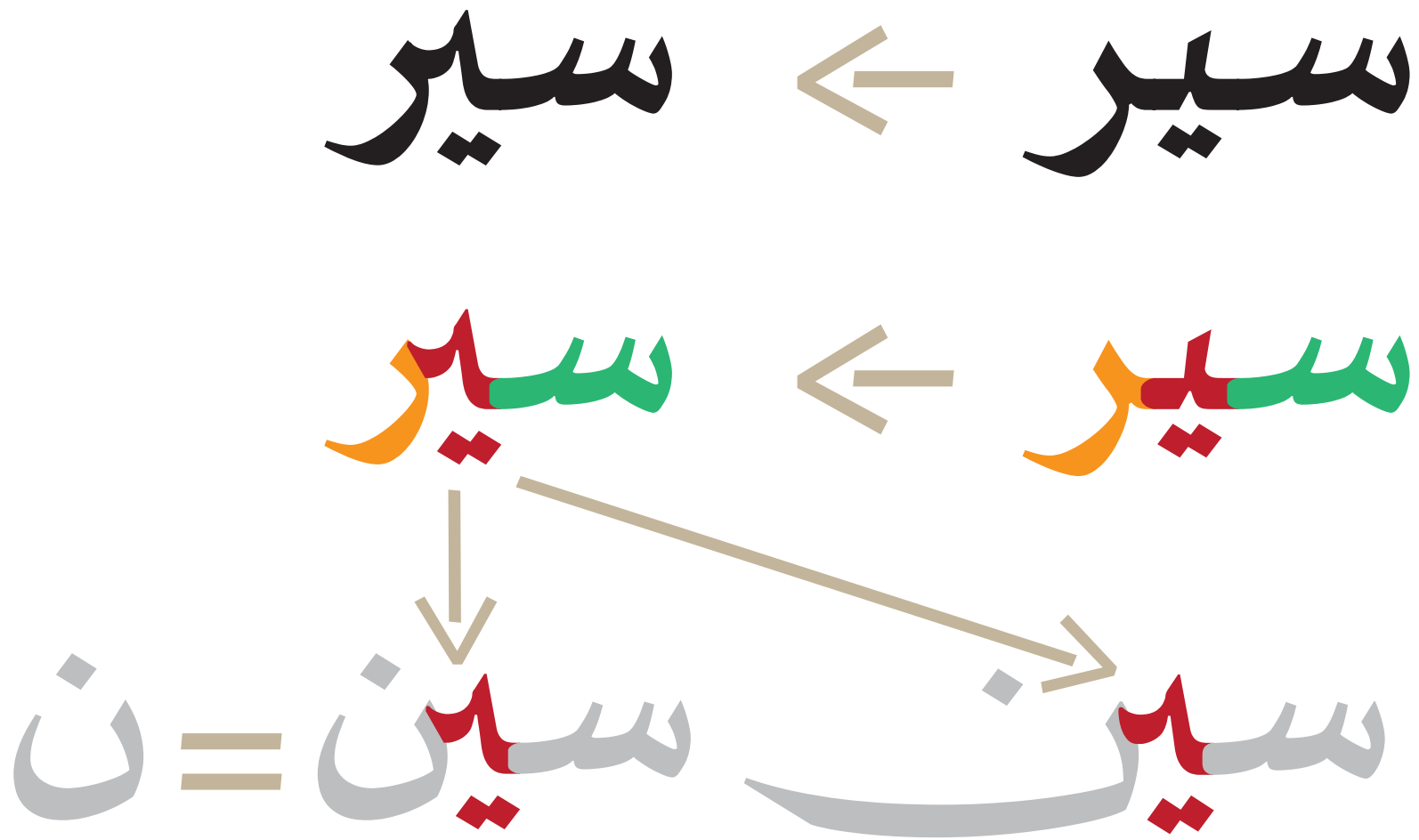


Sakkal Baseet Heavy

Components not Ligatures

- Because of the cursive character of Arabic script it is often necessary to modify the shape of the letters based on the context. For example in Naskh styles, letter Be tooth is raised when it comes before letters Re or Noon.
- There are 33 shapes in the Be tooth group and 14 shapes in the Re group. Using ligatures to cover these we need: 33 x 14 = 462 ligatures.
- Using substitution to cover the same combinations we need: 33 + 14 = 47 glyphs.

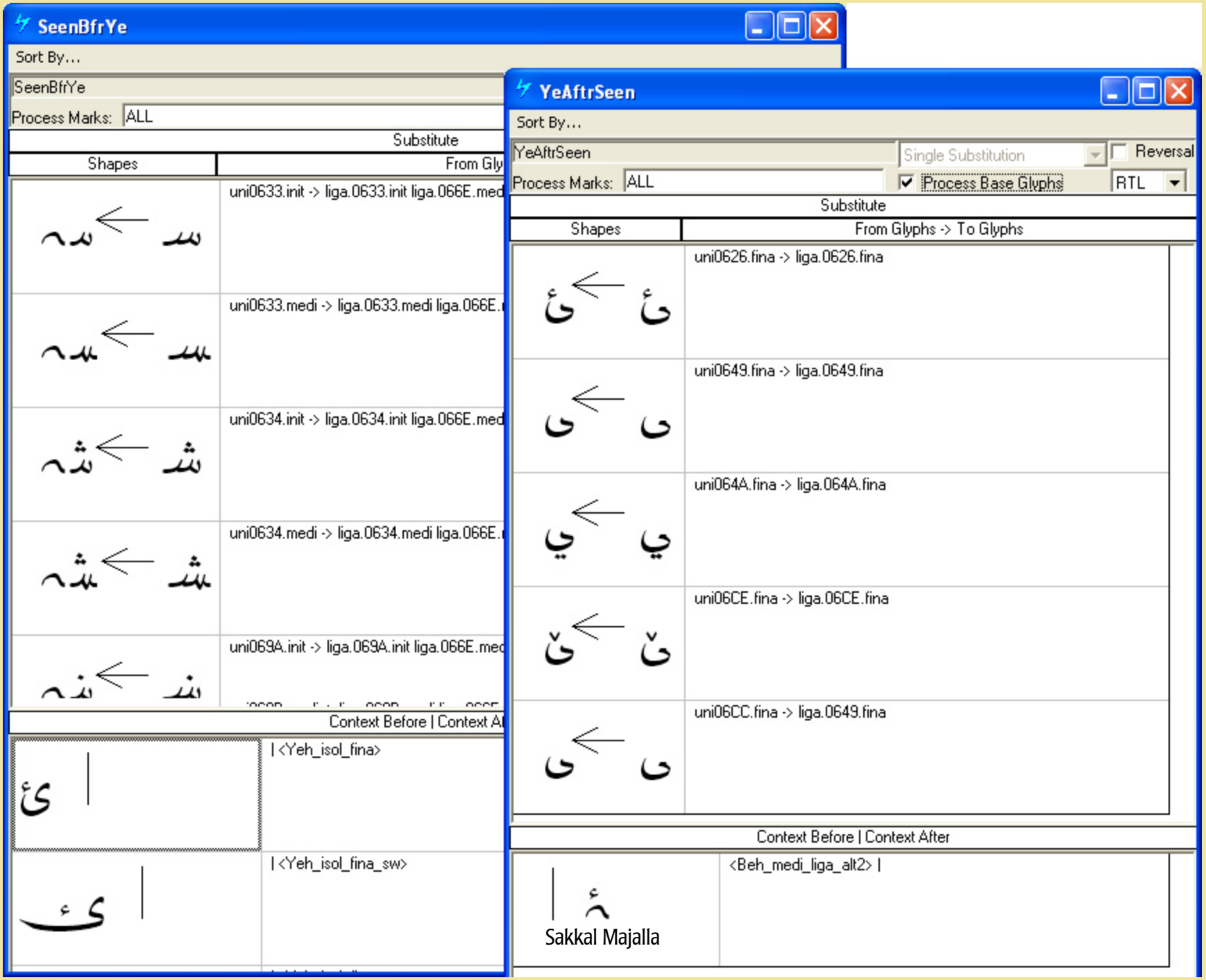

BeBfrRe
Sort By...
BeBfrRe
Process Marks: ALL
Substitute
Shapes
From Glyph
uni0626.medi -> liga.0626.medi
uni0628.medi -> liga.0628.medi
uni062A.medi -> liga.062A.medi
uni062B.medi -> liga.062B.medi
uni0646.medi -> liga.0646.medi
Context Before | Context After
| <Reh_fina_liga>
| <Noon_isol_fina>
ReAftrBe
Sort By...
ReAftrBe
Single Substitution
Reversal
Process Marks: ALL
Process Base Glyphs
RTL
Substitute
Shapes
From Glyphs -> To Glyphs
uni0631.fina -> liga.0631.fina
uni0632.fina -> liga.0632.fina
uni0691.fina -> liga.0691.fina
uni0692.fina -> liga.0692.fina
uni0693.fina -> liga.0693.fina
Context Before | Context After
<Beh_medi_liga> |
Sakkal Majalla

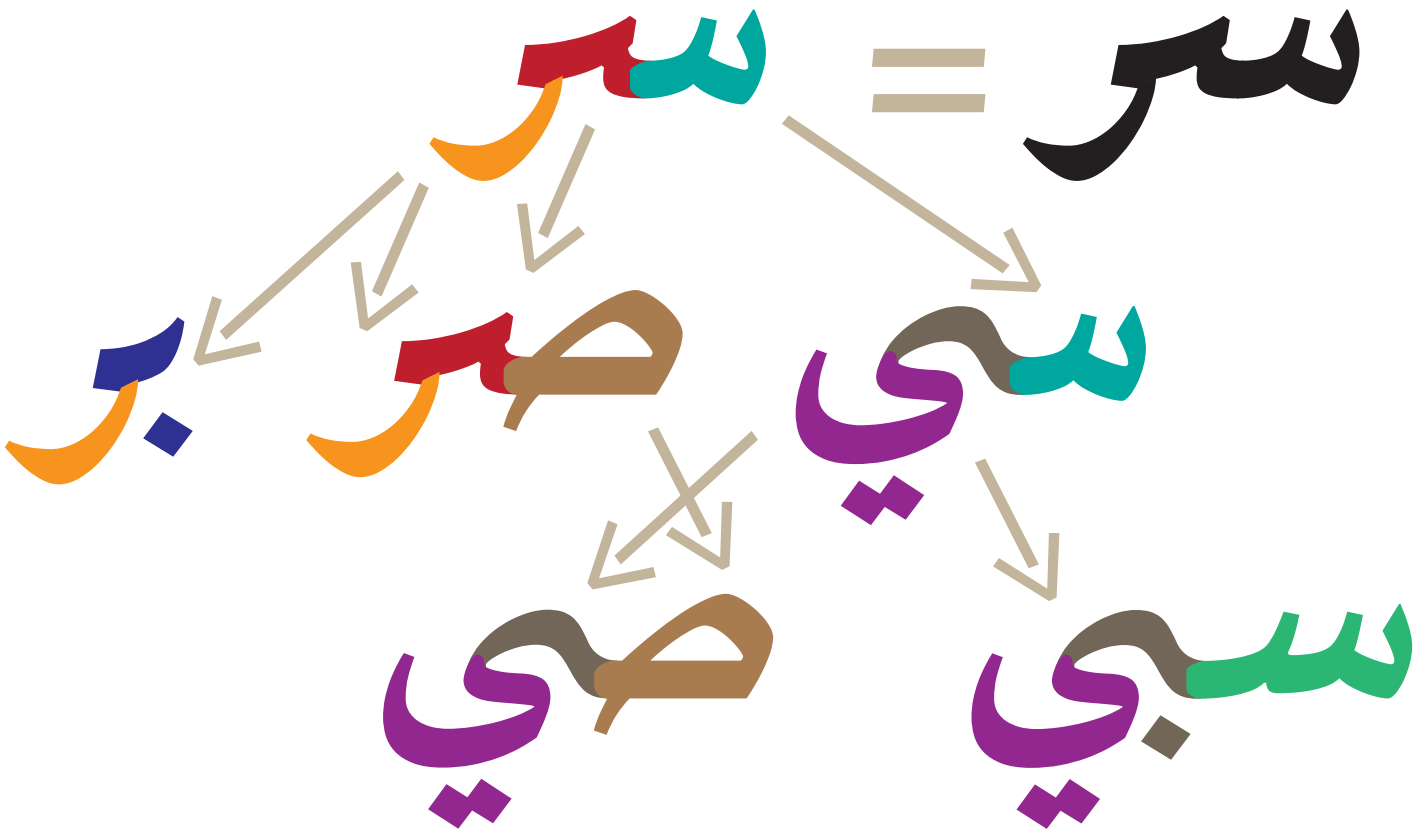


Sakkal Majalla Bold

Components not Ligatures

- In Sakkal Majalla we even designed the Be tooth components to work with the isolated form of Noon so we didn't need to provide a special component for it.
- Another example is Seen before final Ye. Here again the last tooth of the Seen should be raised. We substituted the Seen with two component glyphs which made it possible to use these same components in other substitution lookups as well.
- In these cases, the order of the lookups and the Context Before/After should be carefully planned.


SeenBfrYe
Sort By...
SeenBfrYe
Process Marks: ALL
Substitute
Shapes
From Gly
uni0633.init -> liga.0633.init liga.066E.med
uni0633.medi -> liga.0633.medi liga.066E.
uni0634.init -> liga.0634.init liga.066E.med
uni0634.medi -> liga.0634.medi liga.066E.
uni069A.init -> liga.069A.init liga.066E.med
Context Before | Context A
| <Yeh_isol_fina>
| <Yeh_isol_fina_sw>
YeAftrSeen
Sort By...
YeAftrSeen
Single Substitution
Reversal
Process Marks: ALL
Process Base Glyphs
RTL
Substitute
Shapes
From Glyphs -> To Glyphs
uni0626.fina -> liga.0626.fina
uni0649.fina -> liga.0649.fina
uni064A.fina -> liga.064A.fina
uni06CE.fina -> liga.06CE.fina
uni06CC.fina -> liga.0649.fina
Context Before | Context After
<Beh_medi_liga_alt2> |
Sakkal Majalla



Sakkal Saad Bold

Components not Ligatures

- Another approach to components is to use dot-less skeleton letter shapes, then add the dots to make the different letters with OpenType instructions after decomposing each letter.
- This approach can be applied to single letters, components, or traditional ligatures.

ـ

+

ـ ـ ـ ـ ـ ـ ـ ـ ـ ـ

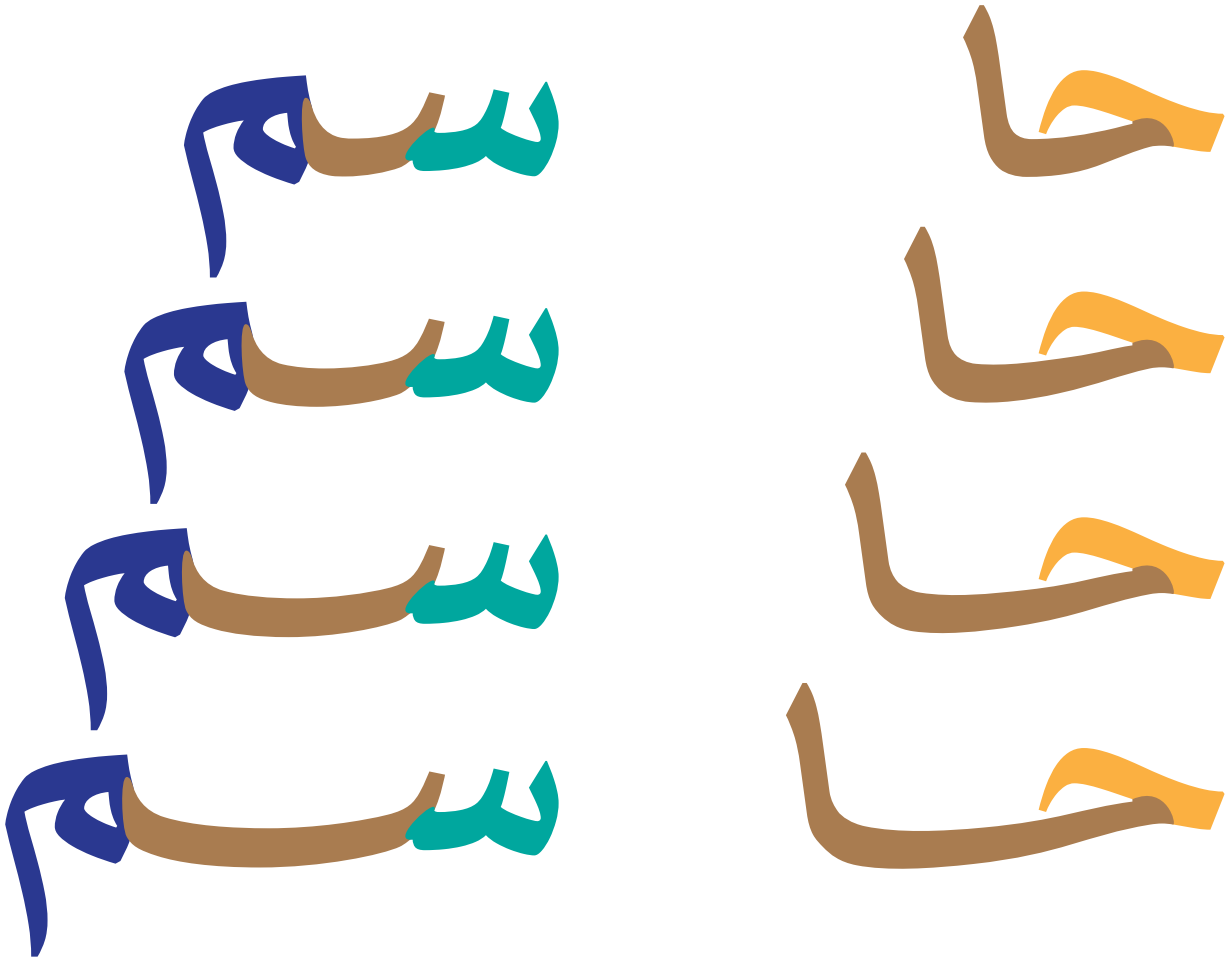
=

ـ ـ ـ ـ ـ ـ ـ ـ ـ ـ

ـ ـ ـ ـ ـ ـ ـ ـ ـ ـ

Components not Ligatures

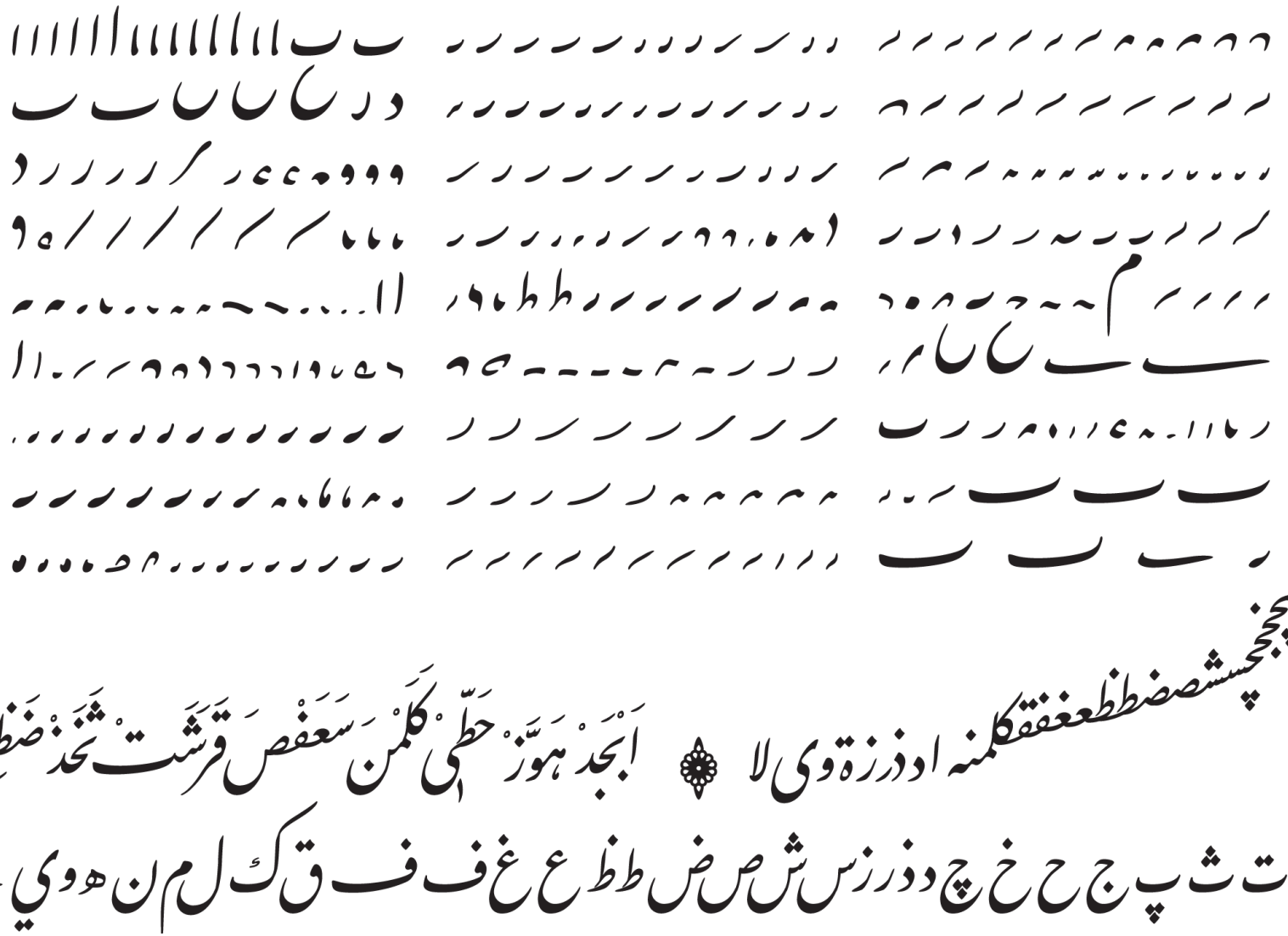
- A useful use of components is for cursive Kashidas, or Tatweel, in cursive style fonts.



Sakkal Saad Bold

Components not Ligatures

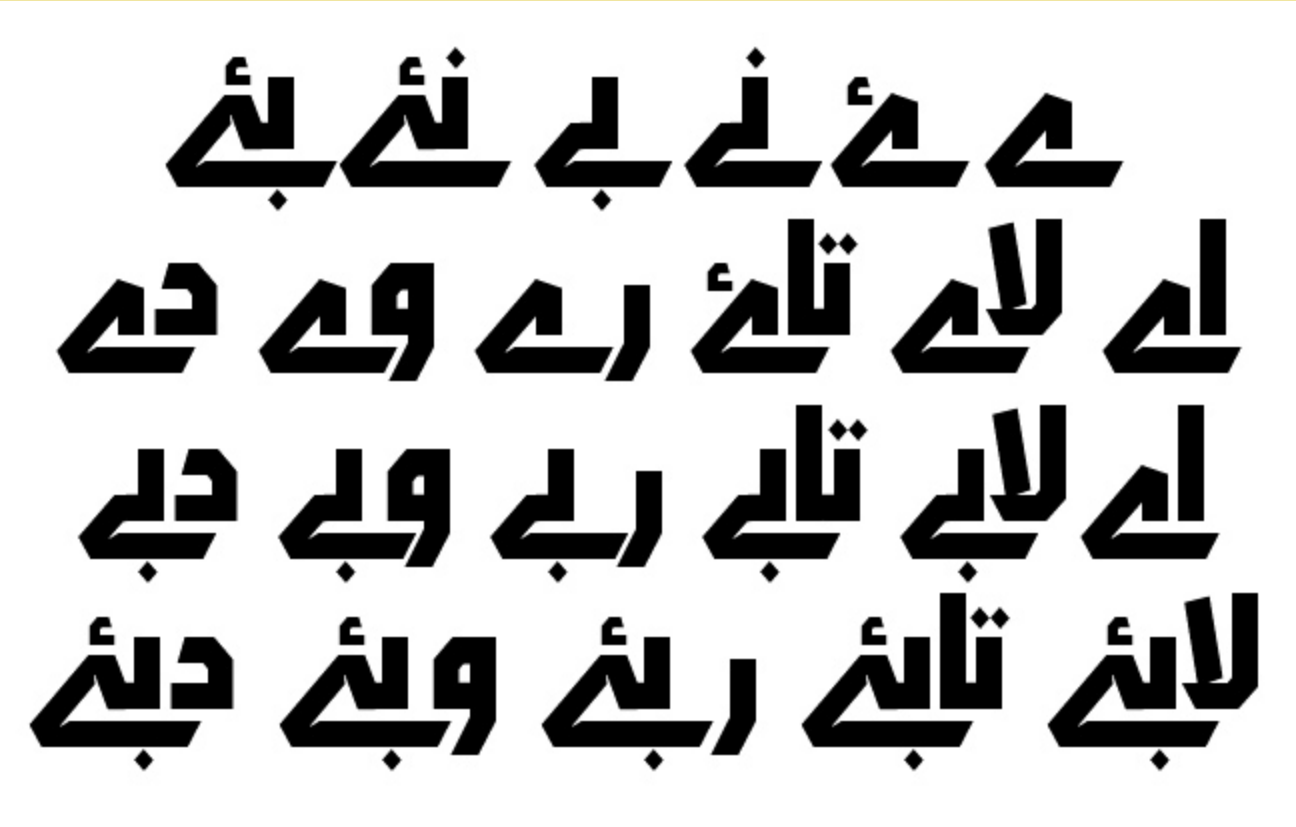
- Deconstructing the glyph shapes into their most basic components will make the font file size extremely small but will require elaborate OpenType programming to put them together.
- Tom Milo and his partners at DecoType are the main proponents of this method for the past 30 years. Rather than using OpenType, they had to develop their proprietary text processing engine, and used it in their application "Tasmeem."



DecoType Nastaliq. Complete glyph set, top, and sample text and letters, bottom

Urdu Ye Barree

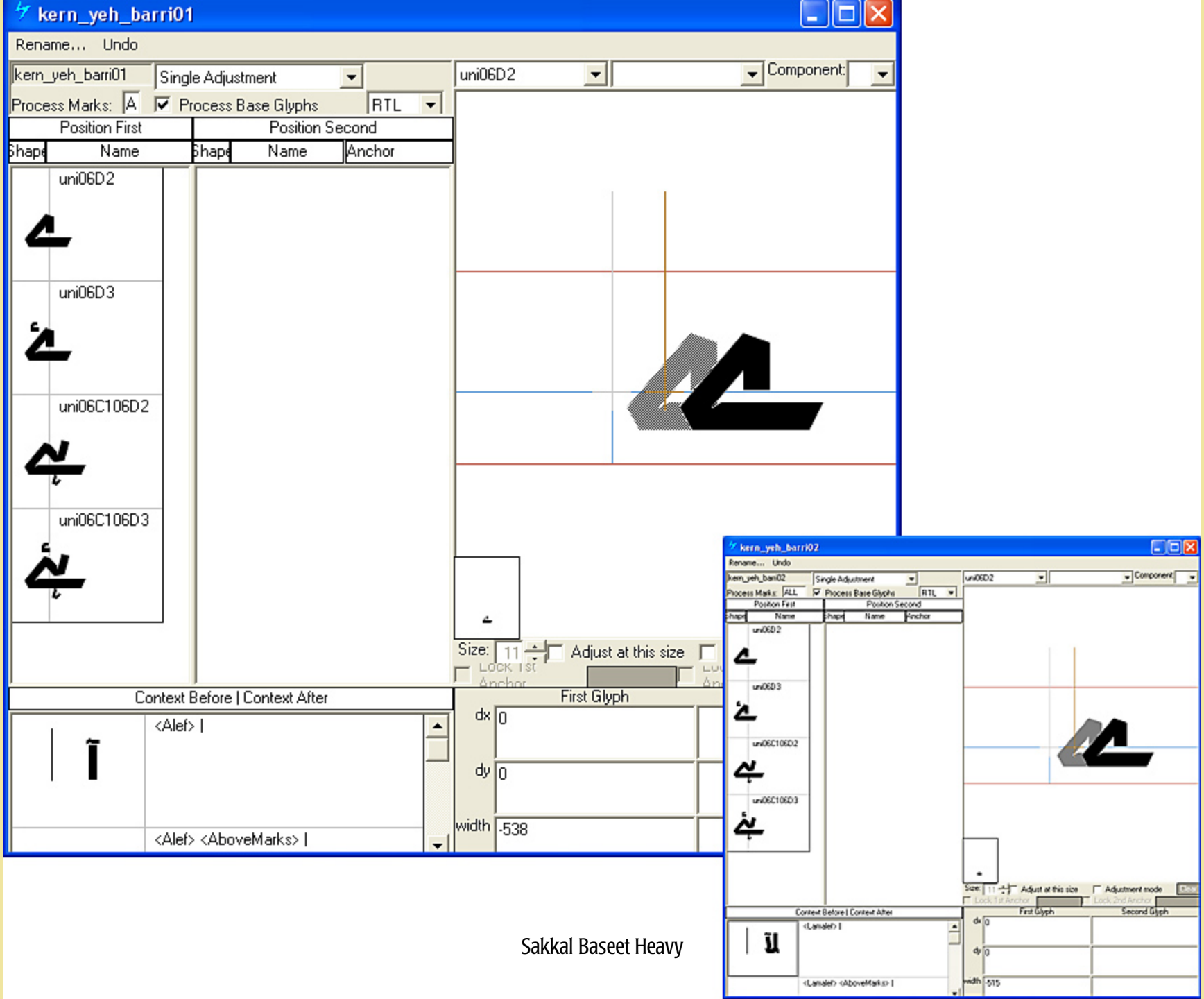
- Ye Barree treatment is one of most challenging conditions in Arabic typography because the tail of the Ye extends below the preceding letters.



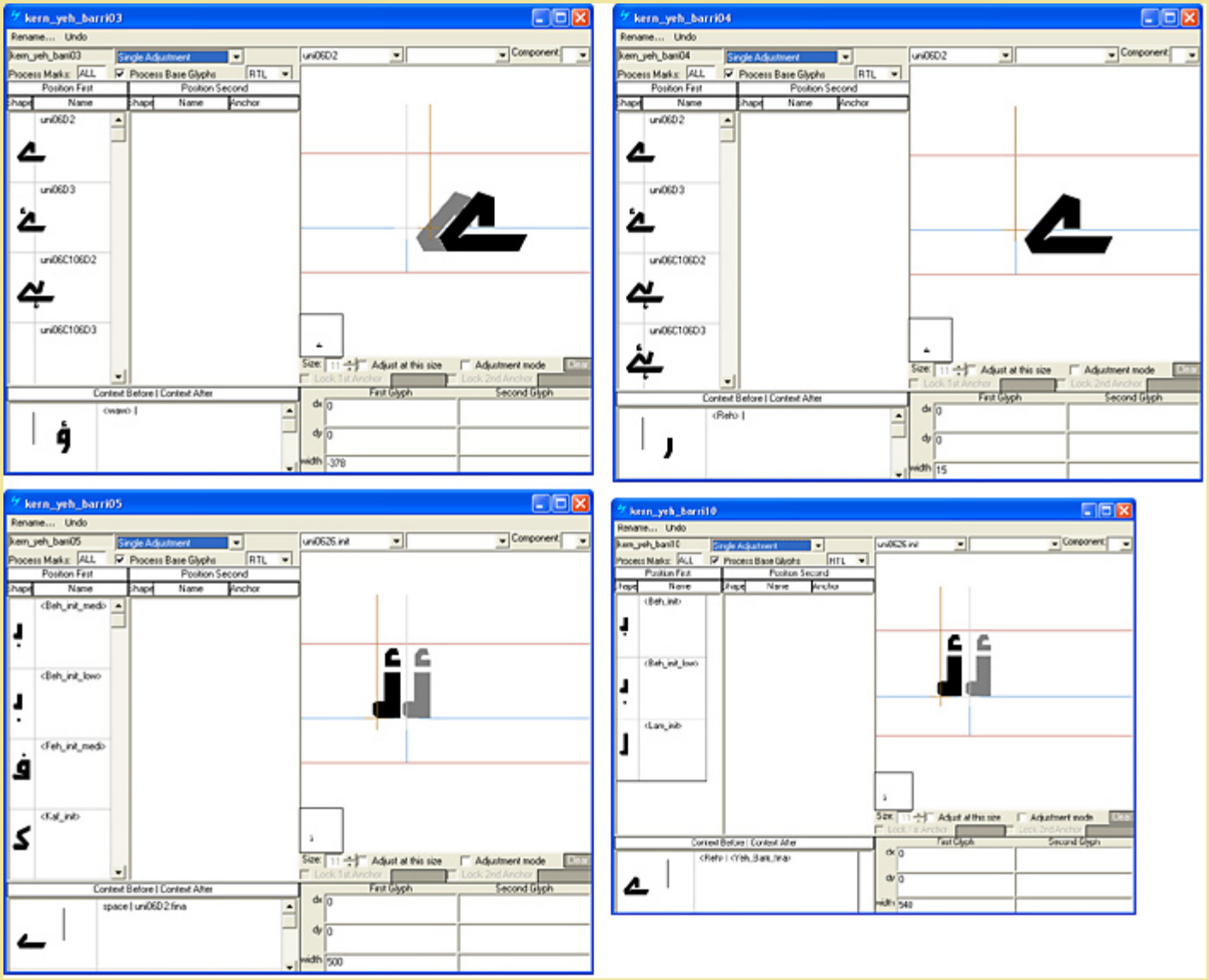
Sakkal Baseet Heavy

Urdu Ye Barree

- Generally, designers extended the Ye Barree tail to the left to avoid this overlap, but this results in ugly typography.
- In “**Sakkal Basset**” we used ten kerning tables to adjust the position of the Ye Barree and/or the letters that precede it to provide a more balanced spacing.
- This example shows that better typography is needed in original, modern typefaces, not only in traditional styles.



Sakkal Baseet Heavy



Sakkal Baseet Heavy

VOLT and OpenType

- Despite some shortcomings in VOLT, it has been an essential tool for type designers who want to take advantage of the possibilities provided by OpenType technology.
- For Arabic script and the many languages that use it, this has been a time of extraordinary progress towards rich and responsive typography for both traditional and modern styles.
- We have only scratched the surface of what is possible. It is clear that the future holds even more promise.

Thank You

Mamoun Sakkal
mamoun@sakkal.com

VOLT DAY

@ Microsoft
June 2, 2005

Updated January 31, 2015