
ORIENTAL MANUSCRIPTS AND NEW INFORMATION TECHNOLOGIES

Thomas Milo

AUTHENTIC ARABIC: A CASE STUDY. RIGHT-TO-LEFT FONT STRUCTURE, FONT DESIGN, AND TYPOGRAPHY*

As the most elaborate of all of right-to-left scripts, Arabic script presents an unusual challenge. The present article focuses on an interaction between text encoding and font technology against the background of understanding the structure of Arabic script. The oldest known Arabic scripts consist of a single layer of ambivalent or multivalent letters. An additional script layer, similar to vowel signs, gradually emerged (with regional variations) to disambiguate letters. Understanding how Arabic script emerged and evolved provides clues both to encoding and rendering issues of Arabic or related scripts. We deal here with a previously underestimated but very powerful aspect of Arabic script which makes it fundamentally *archigraphemic* in structure.

An archigrapheme occurs when the distinction between two or more letters is neutralized. The archigrapheme is a graphic unit that consists of the shared features of neutralized letters minus the features that differentiate them. In the archigraphemic analysis of Arabic script,

vowels and dots are different layers of additional, variable features.

These issues are relevant in the context of Unicode-related discussions, because (i) today the Unicode Standard assumes limited, grapheme-based (i.e. explicit) use of Arabic script; (ii) grapheme and ligature-based legacy technologies have led to misconceptions and inconsistencies both in the code structure and visual rendering of the languages written in Arabic script; (iii) archigraphemic encoding of scripts like Arabic is the key to sophisticated operations on computerized Arabic text corpora and addresses apparent regional and diachronic variation; (iv) the archigraphemic approach is fundamental to proper Arabic font technology; (v) archigraphemic font technology creates optimal conditions for contemporary Arabic font design; (vi) operating systems need to specify the open architecture required to facilitate the optimal technology for rendering a given script, to give the user access to existing and future expert font rendering and layout mechanisms.

Phoneme vs. grapheme

Script terminology is partly inspired by and derived from the linguistic doctrine of *phonology*. Linguistics defines a *phoneme* not as sound, but as a *bundle of distinctive features* in the context of a given language. By analogy the grapheme should not be considered a visible sign, but a *bundle of distinctive features* in the context of a given script.

The linguistic relevance of a feature is established by isolating it from semantically different minimal word pairs, which can be represented as in *Table 1* (see below). This finds a close parallel in the structure of Arabic writing system as represented in *Table 2* (see below).

Table 1

Feature	Labial	Dental	Nasal	Word
Phoneme				
/m/	+	-	+	“map”
/n/	-	+	+	“nap”
The phoneme /m/ differs from the phoneme /n/ in the features of dentality and labiality in a contrastive opposition.				

* This article has grown out of the papers read by the author at MELCOM XXIII, May 2001, St. Petersburg, Russia, and at the 20th International Unicode Conference, Washington, D.C., January 2002.

Table 2

Feature Grapheme	Single dot below	Double dot above	Tooth	Letter
ب	+	-	+	<i>bā'</i>
ت	-	+	+	<i>tā'</i>

The Arabic letter *bā'* ب differs from the Arabic letter *tā'* ت in the features of a single dot below and a double dot above in a contrastive opposition

For the convenience of further representation of the matters under discussion here, we show below 5 different

letters of “*bā'*-class” graphemes whose skeletons are identical, while the attachments are different (see *fig. 1*).

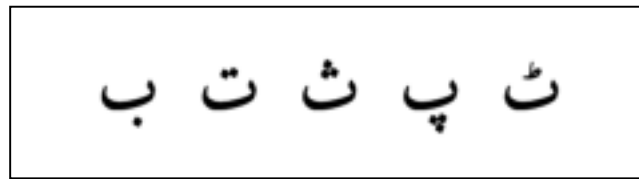


Fig. 1

Allophone vs. allograph

The physical realization of the sound of a phoneme falls outside the scope of linguistics proper; we leave it to the discipline of *phonetics* to analyze and describe it. The sound of a phoneme has many subtle context-determined variations that do not affect the linguistic meaning and,

therefore, escape the native speaker — the *allophones*. For the phoneme /n/ they can be illustrated as represented in *Table 3*. There are also subtle variations in shape that usually escape the attention of a non-sophisticated reader. See *Table 4*.

Table 3

Phoneme	Contextual positions of allophones
/n/	[n- -n- -n]

A phoneme can occur in the initial (n-), medial (-n-) and final (-n) position. In each position, there are variations of the actual sound caused by modulation as a result of the surrounding sounds influence — allophones.

Table 4

Grapheme	Contextual positions of allographs
Arabic letter <i>bā'</i>	[ب به اب]

The contextual positions — initial, middle and final — are the allographic categories. The actual allograph is the result of interaction with the allographs of any adjacent graphemes.

Most simplified fonts have only *one* glyph of each position to cover allographs. Legacy typography incorporates

a small, random selection of additional allographs in “nostalgic” *ligatures* (see *fig. 2*).

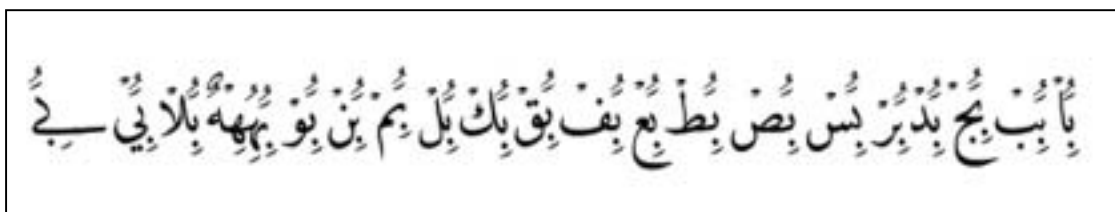


Fig. 2 A selection of “*bā'*-class” allographs in the initial position. The theme letter (*bā'* in this example) is surrounded by a double set of vowels and followed by a parade of final forms (generated by DecoType Arabic Calligraphic Engine).

Interestingly, there is no traditional pattern for listing the middle forms [1]. In the example below, the preceding selection of “*bā*’-class” allographs is expanded to show

a small selection of these allographs in the middle position (see *fig. 3*).

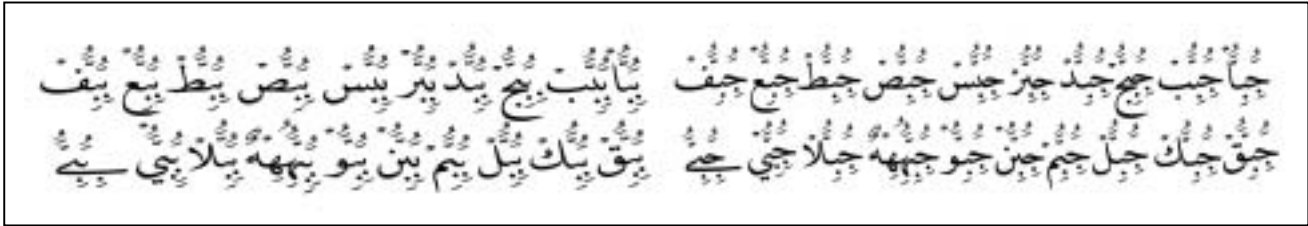


Fig. 3 The two examples of “*bā*’-class” allographs in the middle position in a quasi-traditional presentation showing all final forms. The block on the right shows an initial *bā*’, while on the left — initial *jīm* (generated by DecoType Arabic Calligraphic Engine).

The visual realization of graphemes falls outside the scope of Unicode. It is the field of expert technologies to

handle the allographic level of the Arabic script and to create the right conditions for professional type design.

Archiphoneme vs. archigrapheme

In the sound system of Classical Greek — and that of most languages — there are no minimal word pairs with the opposition /n/:/m/ when these phonemes are followed by

/b/ or /d/. In fact, while /mb/ and /nd/ exist, /md/ and /nb/ are ruled out [2]. See *fig. 4*.

+ Dental		+ Labial	
ντ	νδ	μπ	μβ
νθ	νζ	μφ	

Fig. 4 Combinations of /n/ or /m/ with following dental or labial consonant (classical Greek).

The functional difference between these phonemes disappears and results in a new phenomenon — *archiphoneme*

(symbolized with a capital letter of one of the neutralized phonemes). See *Table 5*.

Table 5

Feature	Labial	Dental	Nasal	Example
Archiphoneme				
/N/	(+)	(-)	+	<i>embryonic</i>
/N/	(-)	(+)	+	<i>endemic</i>

Archiphoneme is a phonological concept that consists of the shared features of neutralized phonemes minus the features that differentiate them [3].

In a large corpus of historical Arabic texts, the distinctive features is a rare thing, since they were employed

rather sparingly. This enables us to make the analogy as presented in *Table 6*.

Table 6

Feature	Single dot below	Double dot above	Tooth	Example
Archigrapheme				
◌ِ	(+)	(-)	+	<i>bā</i> ’? <i>tā</i> ’?
◌َ	(-)	(+)	+	<i>tā</i> ’? <i>bā</i> ’?

Archigrapheme is a graphic unit that consists of the shared features of neutralized letters minus the features that differentiate them. In this analysis of Arabic script, vowels and dots are different layers of additional, variable features.

Many important historical texts are known only in a “defective”, i.e. archigraphemic script. Even if an old manuscript in *scriptio plena* exists, it often derives from an archigraphemic original, which implies that the layers of secondary script are later interpretations. These additional layers — both vowels *and* dots — are the ones most vul-

nerable to scribal errors. In fact, these documents are, strictly speaking, only truly original on the archigraphemic level. An academic analysis of the computerized versions of such corpora is frustrated by the present graphemic structure of Arabic in Unicode. Alternative archigraphemic encoding with roundtrip compatibility would be ideal.

Font structure's inadvertently effect on data structure

The adaptation of Arabic script to the typewriter was the ultimate step in the process of simplifying its morphology. It is a classical case of quality sacrificing to minimize design effort. To freeze this fluid writing system on just forty-four keys, it was stripped of all ligatures but one. *Alif* follows *lām* in such a way that this particular sequence of

two letters cannot be dissected. As a result, *one* key represents a curious ligature. With a bit of imagination, the *lām-alif* key is more than a permanent carry-over from typography. It can be regarded as a metaphor of the resilience of the Islamic writing culture against mechanical maltreatment, since in Arabic it means “no”! See *figs. 5 and 6*.



Fig. 5



Fig. 6

With the evolution of the keyboard into a data-entry tool, old typing habits create an interesting problem. In order to be linguistically consistent with the consonant-

plus-vowel structure, attachments should follow the governing letter directly (see *figs. 7, 8*).

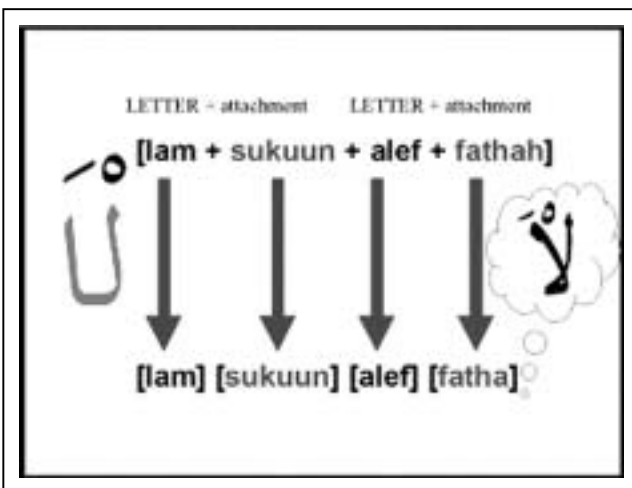


Fig. 7

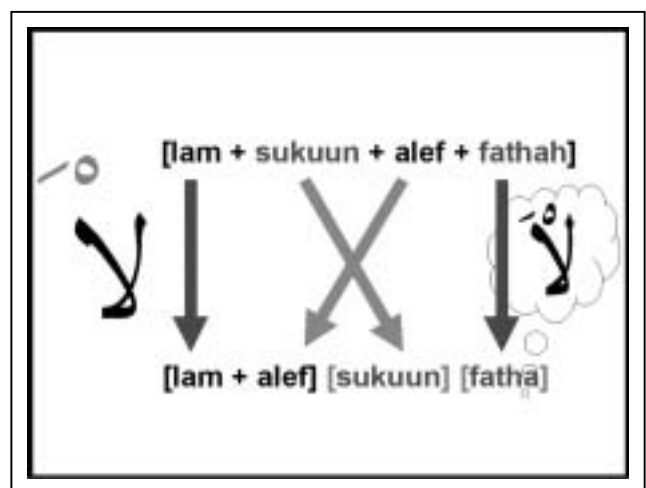


Fig. 8

However, today the widely used table-driven Arabic script does not generate ligatures of letter groups when they are separated by attachments. Therefore, in the case of *lām-alif*, to achieve the correct visual result the writer is forced

to rearrange the data in a manner that is graphemically incorrect — or just to forget about the diacritics. This is just one example of how defective font technology has created chaos in the world of computerized Arabic.

Data structure's inadvertent effect on font structure

In the post-1920s Arabic typography, a related problem with attachments can be observed. The type-setter has no means to insert attachments to letters that are part of a ligature, so he replaces them with individual “typewriter”

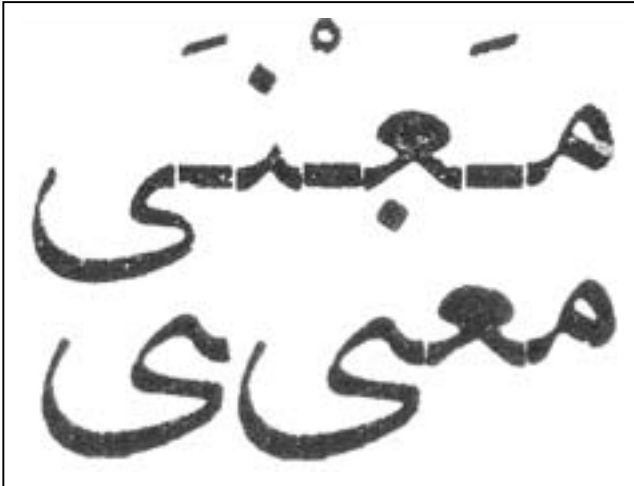


Fig. 9 English metal font (1950s) shows crude insertion of vowels with sacrificing a ligature (shown separately as well).

In computing, the erroneous classification of ligatures as *optional* leads to shaping algorithms that allow falling back on “typewriting” when inserting vowels, with comical effects as described above. In properly designed Arabic font technology, the attachments would not influence the



Fig. 11

At this point, it must be stressed that this type of defect hampers *all* sophisticated fonts that were conceived to function with ligatures and full vocalization. Technology has wreaked havoc under Arabic type instead of facilitating it. Even innova-

glyphs in order to fit in the attachments, often “camouflaged” by an extra-carrier line. Mechanically, he has no other option but to sacrifice the typographically correct ligatures, and he assumes this is aesthetically acceptable.



Fig. 10 Arabic computer typefaces accommodate for vowels by sacrificing ligatures (dimmed in the background).

main script. To illustrate it, we present here the images, which were generated with the aid of the DecoType Arabic Calligraphic Engine technology, demonstrating the attaching of distinctive dots and vowel markers without affecting the skeleton text. See *figs. 11 and 12*.



Fig. 12

tive design, simplification and restructuring of Arabic script as a conscious cultural choice is caught in a straightjacket of technical shortcomings. Operating systems need to provide the open architecture for expert systems to deal with such issues.

Grapheme-based legacy Arabic typesetting technology

Why vowels should affect the structure of the graphic skeleton can be understood from analysing the mock-up below. It shows *grapheme*-based Arabic typographic technology: it treats dot-attachments as integral part of the let-

ter [4]. Interestingly, remnants of an earlier *archigrapheme*-based technology can be seen in this design as well: the ligature on the right allows the dots to be attached separately.

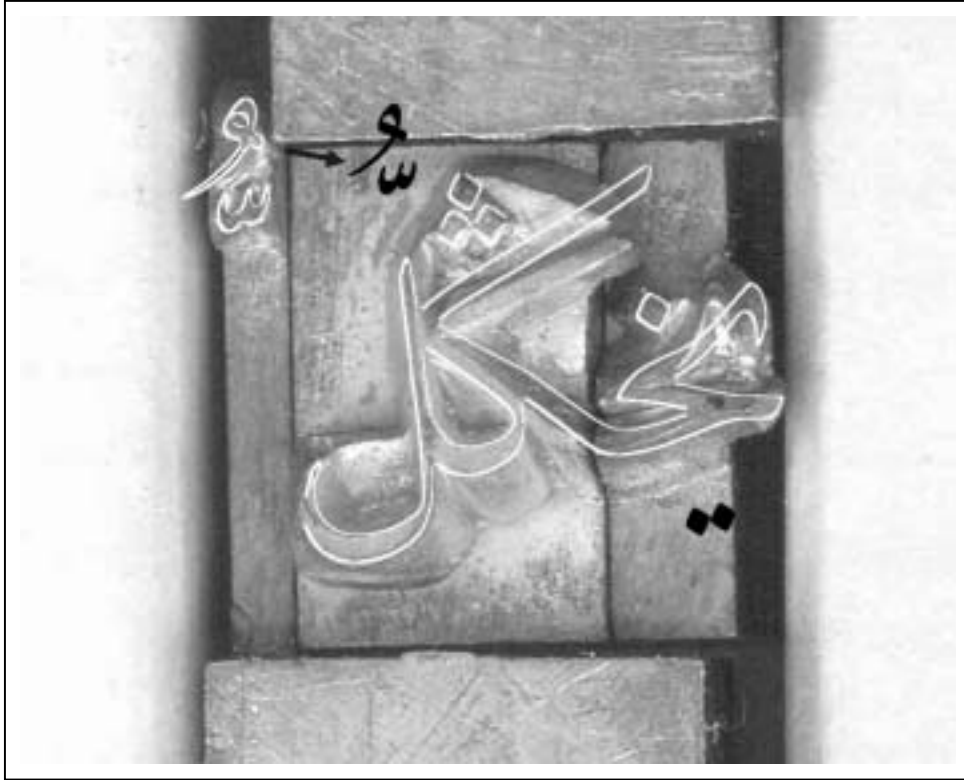


Fig. 13 The two main castings contain built-in attachments (one has a single and the other a triple upper dot). The ligature on the right is designed to allow an attachment to be packed under it, placing attachments over or under its extending pointed shape, e.g. *yā'-khā'*- (when two dots are added as in the example). The letter block on the left represents two attachments united in a ligature. This particular metal construction positions these attachments significantly away from the last letter. However, a natural place for them is above the last letter — *lām* with *shadda* and *ḍamma*. Because of the graphemic structure of the font, it cannot deal correctly with Arabic script.

The mother of Arabic typography was based on archigraphemes

The Ottoman *naskh* (*nesih* in Modern Turkish) definitely guided all Middle Eastern efforts in typography. In the 1860s, the Armenian typographer and Ottoman citizen, Ohannis Mühendis-oğlu (1801—1876) [5], after many at-

tempts, succeeded in reproducing this script in a way that met the demanding standards of the Islamic calligraphic tradition [6]. His sublime approach to typographic solutions appears to have been fully *archigrapheme*-based.

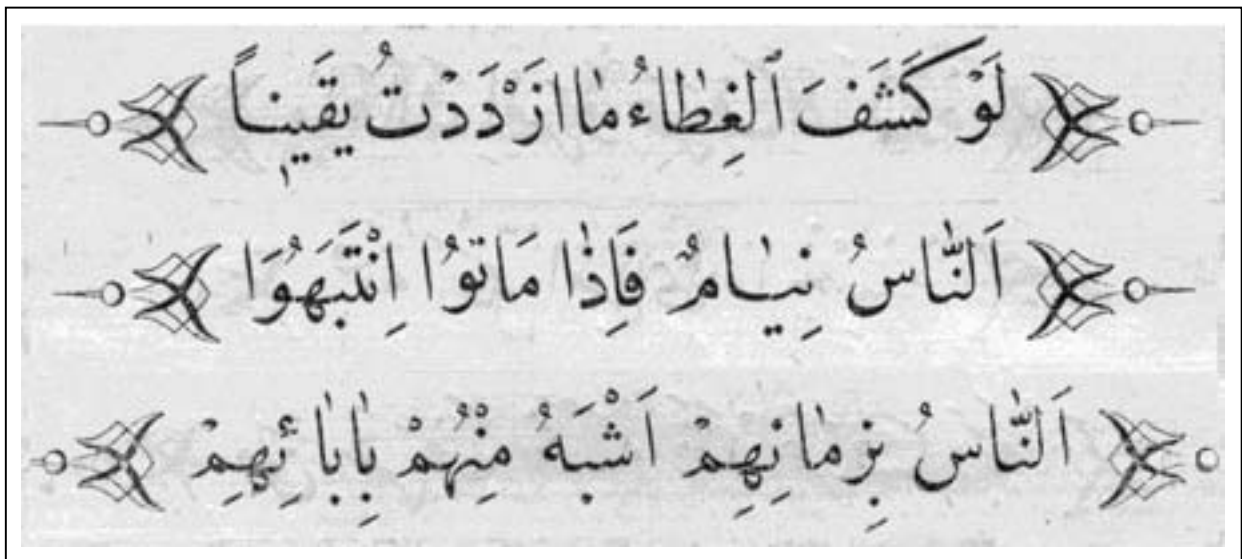


Fig. 14 Arabic phrases, typeset in metal, showing integral coverage of the Arabic script morphology and correct placement of the attachments.

The previous illustration (see *fig. 14*) shows close-ups from brilliant typesetting by Mühendis-oğlu in the *Yeni*

Hurufat [7] in the three main languages of the Ottoman Islamic world — Arabic, Persian and Turkish.



Fig. 15 Each Arabic phrase is followed by a Persian translation and an Ottoman Turkish explanation.



Fig. 16 The same trilingual text with our indication of the language used.

Any given language has its own distinct pattern of sound combinations, so when an Arabic font is used for two extra- (totally different) languages, it exposes more of the structure behind glyph combining and ligature using than any monolingual typography. Languages do not just differ in respect of phoneme inventory; they also differ in phoneme usage. *Fig. 17* (see below) shows that “permissible” combinations of a certain

language [8] produce a unique “fingerprint”. Another example can be seen in English which happens to differ from Greek in the distribution pattern of the cluster /ps/: in the borrowing *ellipsis*, /ps/ can be matched with English phonemes, but not in the word *psyche*. If English had been spelled morphophonologically — like Arabic — we would have never seen the letter group *ps-* in the initial position: *sykee*.

The phonological system																	
Bulgarian Initial Clusters of two Consonants																	
	p	f	t	s	c	č	š	k	x	b	d	z	ž	g	v	m	n
p				×				×									
p'				×													
f				×													
t	×	×		×				×									
t'				×				×									
s	×	×								×							
s'		×															
c				×													
c'				×													
č	×	×															
š	×																
k			×	×				×									
k'				×				×									
x				×													

Fig. 17

Rather than to attempt to create his own version of Arabic script, Mühendis-oğlu aimed to model his typography on the handwriting of *kaziasker* (chief military judge) Mustafa İzzet Efendi (1801—1876), one of the high-ranked Ottoman officials. İzzet Efendi, a man of great authority, was also a composer of Ottoman classical music and the leading calligrapher of his time. Among his numerous calligraphic works are inscriptions inside the Aya Sofya Mosque, the main sanctuary of Istanbul. This lofty man certainly was not the type to be involved in *type design*, and it can be ruled out that the craftsman and the calligrapher ever met. Below, we present the *meşk murakka'i*, or writing exercise, by İzzet Efendi, elaborating the shapes of the letter *kāf*, in the *naskh* style, the artistic equivalent of the *étude* of the Western musical tradition (see *fig. 18*), and the interior of the Aya Sofya Mosque in Istanbul, decorated with circular calligraphic inscriptions by İzzet Efendi, containing the names of the Caliphs ‘Uthmān and Abū Bakr (see *fig. 19*).

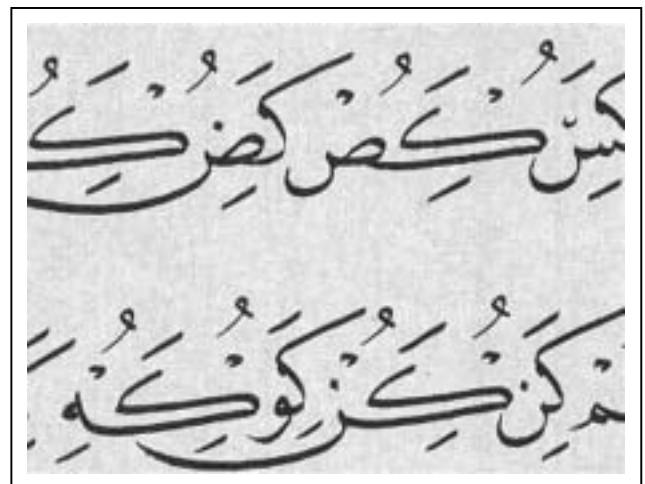


Fig. 18



Fig. 19 Fig. 20 Mühendis-oğlu

Mühendis-oğlu's adaptation of İzzet Efendi's calligraphy is the starting point of all later Arabic *naskh* typefaces. The font was graphically extremely sophisticated as it was designed to follow all the *allographic* rules of *naskh* in the tradition of copyists, the professional book producers before the advent of typography. The essential feature is that it deals with both dot and vowel attachments as separate horizontal layers above and below the main script. In other

words, the design was *archigraphemic*. However, the seeds of decay are already present in this 40-page booklet. The initial pages immaculately implement every rule with the correct glyph. As the page numbers go up, so goes the number of calligraphic typos: the zenith of Arabic typography stands at the beginning of erosion rather than evolution of *naskh* script [9]. This is an extremely good design, but it should have had a computer program to support it!

Allographic decay

All contemporary fonts show considerable simplification that leads to a drastic change in the appearance of Arabic script. Below, step by step, using a minimal selection of glyphs from the *Yeni Hurufat*, the road to the currently widespread font type is reconstructed. The “*bā*-class” had been provided with a special initial curve assimilated to the archigrapheme *qāf/fā* (see fig. 21). We show also (see fig. 22) a specialised “*bā*-class” allograph which occurs only when it is preceded by certain inverted “*bā*-class” allographs. In simplified typography it is used as a generic middle form. Finally, a curved

“*bā*-class” allographs are represented in fig. 23, in a word composed by Mühendis-oğlu as (i) designed (above) and (ii) as composed with out-of-context borrowings from the same design. The result is modern typography! Even the best-designed font becomes pedestrian when generic forms are used. In fig. 24, *thā*-*rā*’ is shown in the context position (above) and when composed with out-of-context elements. It should be noted that the combination of new generic initial and middle forms cannot be used without breaking the rules of the traditional allographic system (see figs. 25—28).

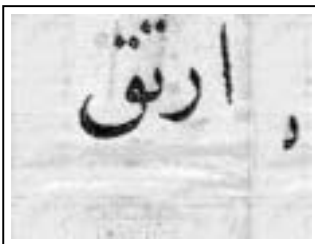


Fig. 21



Fig. 22

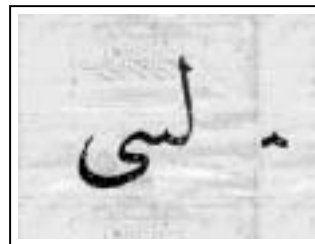


Fig. 23



Fig. 24

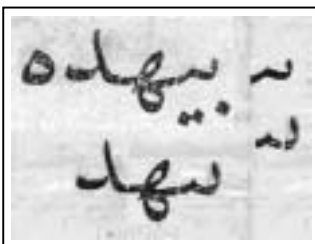


Fig. 25

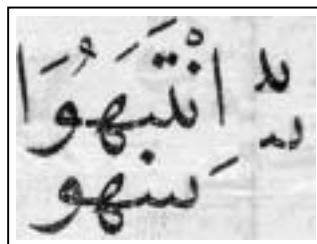


Fig. 26

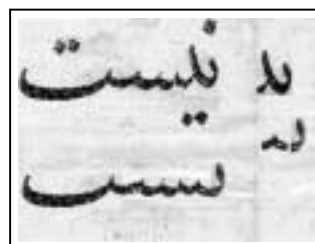


Fig. 27

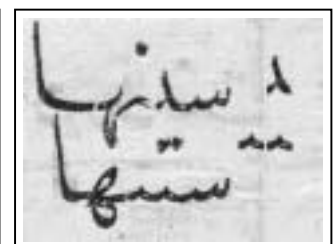


Fig. 28

An interesting example of the lost know-how is two “*bā*’-class” allographs’ occurring between dissimilar letters; for this particular case Mühendis-oğlu cut a special double-pointed curve ligature. However, this typesetting method must have been very tiring to be handled in the booklet, judging by the fact that Mühendis-oğlu applied the correct form only twice. As a matter of fact, even his sublime *naskh* design cannot implement this rule in all contextual situations, because in a metal font boundaries are necessa-

ily on the graphemic level, while the underlying calligraphic mechanism operates on the level of pen strokes regardless of the graphemic status of the larger unit they build. The illustrations below (see *figs. 29–31*) show a special ligature for *-bā*’-*bā*’- in the middle position (modern fonts glyph sets do not include this feature). The double curve ligature crosses grapheme boundaries: one part belongs to the (archi-)grapheme *bā*’ and the other part belongs to the following or preceding (archi-)grapheme.

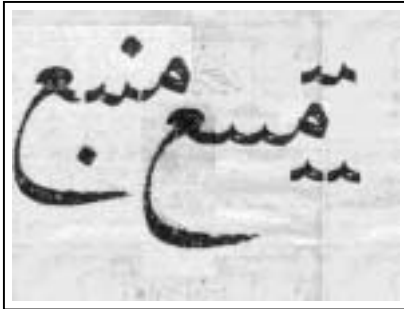


Fig. 29

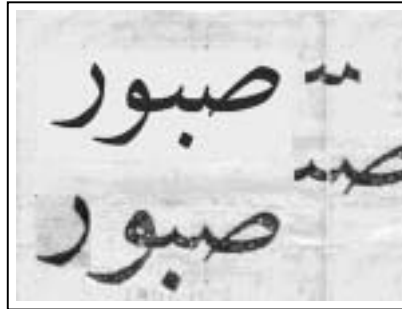


Fig. 30



Fig. 31

The use of the archigrapheme concept in font technology

The Operating Systems provide new font technologies in the wake of the emerging Unicode Standard. In the case of Arabic, they can be put to good use for building grapheme-based fonts for optimal Unicode coverage. However, the Unicode Standard has no provision for archigraphemic Arabic yet. Nevertheless, this fundamental structure of Arabic script can be exploited at least in the design phase: one can build an automated Arabic-specific font tool.

FontLab, the most up-to-date font-designing software, has been adapted jointly by its designers and Dec oType to produce simple OpenType fonts for Adobe InDesign and

WindowsXP. This tool also provides an efficient interface to build legacy style ligatures of up to 4 graphemes.

The archigraphemic structure implies that Arabic graphemes share many structural elements. This phenomenon was exploited here to the maximum. We show below the examples of using “*bā*’-class” graphemes: the skeletons (archigraphemes) are *identical* and the attachments (distinctive features) are *different*. Some letters share the same sub-letter element (see *fig. 32*) and some — the same attachment (see *fig. 33*). This forms the basis of the automation process.

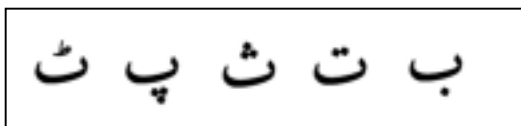


Fig. 32

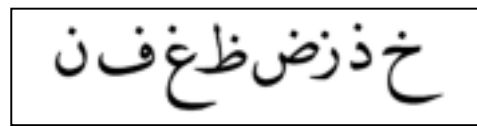


Fig. 33

If the repetitive nature of Arabic writing is exploited, only the outline paths of a limited set of sub-letter elements need to be drawn. The printable glyphs are composed by

references to these base glyphs (see *figs. 34–43*). The practical result is an extremely small font easy to be designed and maintained (see *fig. 44*).



Fig. 34



Fig. 35

*Fig. 36**Fig. 37**Fig. 38**Fig. 39**Fig. 40**Fig. 41*

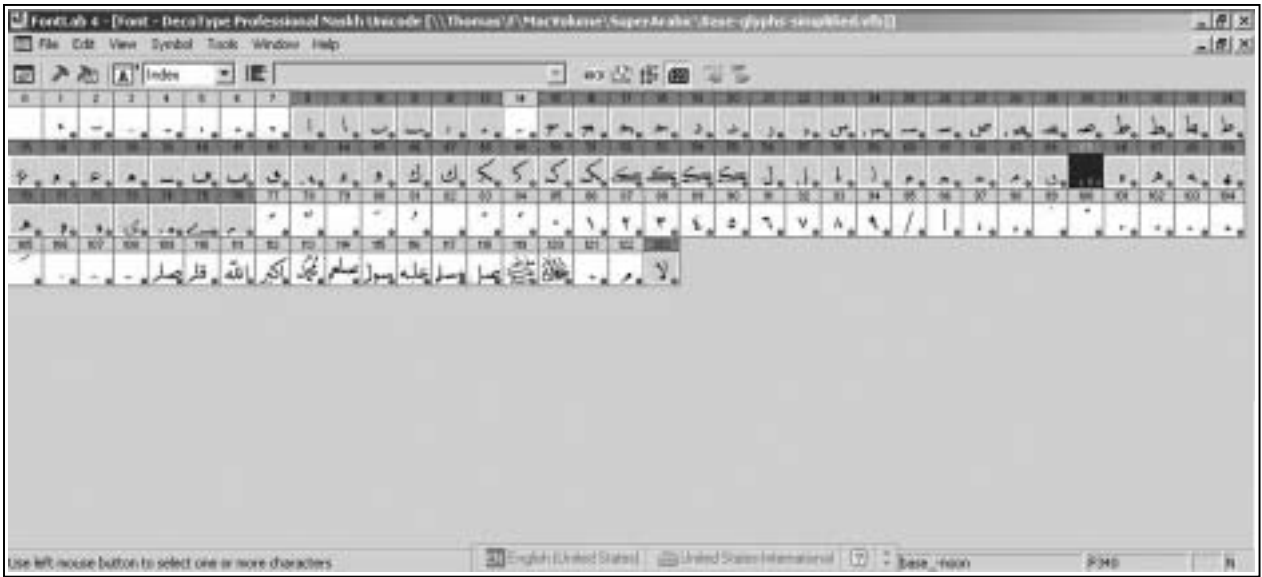


Fig. 45

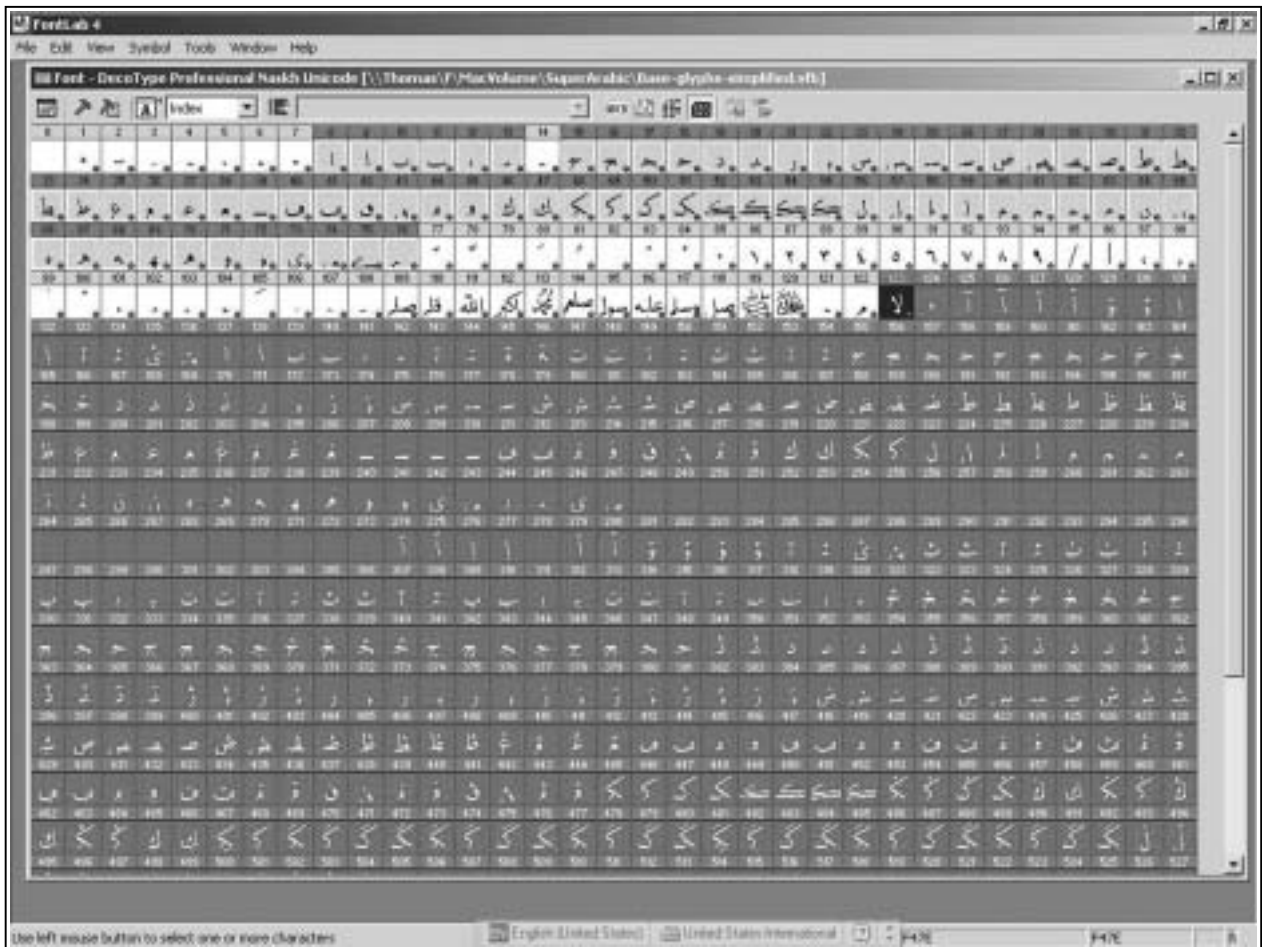


Fig. 46

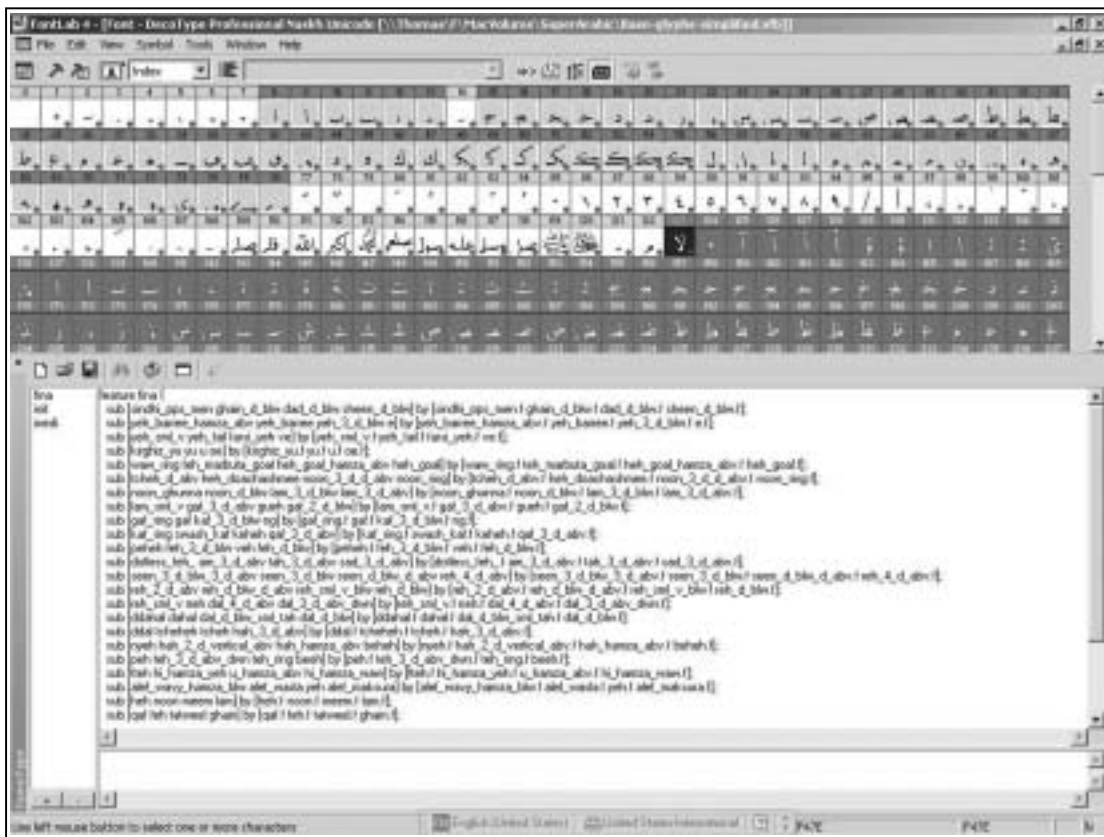


Fig. 47

The method described above shows just one way of benefiting from the archigraphemic structure of Arabic script. The drawback of this table-driven paradigm is that it is archigraphemic only in the design phase.

To conclude, what is required in Unicode is a variant character-glyph model to handle these issues on the level of the Operating System. But it is not just that. For the

end-user it would be a major improvement if Operating Systems in general facilitate the use of the optimal script system for a given script. This means the ability to switch between different Font Rendering and Layout mechanisms. An example of such modularity is Apples Open Font Architecture (OFA).

Notes

1. For modern tables, see *Writing Arabic, a Practical Introduction to Ruq'ah Script*, ed. T. F. Mitchell (Oxford, 1953). These tables cover only groups of two and three base letters, i.e. a fraction of the total required.
2. W. Brandenstein, *Griechische Sprachwissenschaft I. Einleitung, Lautsystem, Etymologie* (Berlin, 1954), § 50.
3. A. Cohen, C. L. Ebeling, K. Fokkema, A. G. F. van Holk, "Fonologie van het Nederlands en het Fries", *Gravanhage* (1971), p. 49.
4. The metal elements were once part of an Arabic font produced by the Dutch Tetterode company, from which I managed to salvage a few types. On the basis of the available forms, the example, therefore, has to be random. I mirrored it for the sake of comparison.
5. Mühendis-oğlu is the Turkish name (lit. "son of the land surveyor (or engineer)") of the Armenian typographer. We also find his name in its Ottoman-Persian (Mühendis-zade) and Armenian (Mühendisyan) form.
6. An *arzuhal* (petition) to the Ottoman Sultan dating by A.H. 1283 / A.D. 1865 came across me in 1983. Its author, Mühendis-oğlu, announces that for the first time a valid *naskh* typeface was designed by him. He describes how he used the handwriting of the late *şeyhülhattatin* (leading calligrapher), Mustafa İzzet Efendi, to accomplish this historical achievement. Uğur Derman, the leading specialist in Ottoman calligraphy, reports corroborating evidence to the Turkish Librarians' Association. See his "Yazı sanatının eski matbaacılığımıza akisleri", in *Türk Kütüphaneciler Derneği Basım ve Yayıncılığımızın 250. yılı Bilimsel Toplantısı. 10–11 Aralık 1979, Ankara. Bildiriler* (Ankara, 1980), pp. 97–118. In this essay, he also mentions the advanced *ta'liq* typefaces designed by Mühendis-oğlu as early as 1840s. In spring 2001, I discovered two of only three surviving books ever printed in Mühendis-oğlu's *ta'liq*.
7. According to the colophon, it was printed in Istanbul in 1869/70. In spring 2001, I made the sensational chance discovery of this rare book printed in the very same *naskh* typeface as the petition of 1865 (see n. 5).
8. Bulgarian consonant clusters at the beginning of a word were borrowed from H. I. Aronson, *Bulgarian Inflexional Morphology* (the Hague, 1968).
9. *Naskh, thulth, (naskh-i)talīq* and *ruq'a* (or *riq'a*) scripts are governed by well-organised and logical morphological rules, the knowledge of which is rare among typographers and type designers today. Even in the *naskh* typesetting of his first 1865 petition, Mühendis-oğlu makes two composition errors.