



HAL
open science

Sabra, a Syriac TeX System

Yannis Haralambous

► **To cite this version:**

Yannis Haralambous. Sabra, a Syriac TeX System. SyrCOM-95, First International Forum on Syriac Computing, Jun 1995, Washington, United States. pp.3-23. hal-02101581

HAL Id: hal-02101581

<https://hal.science/hal-02101581>

Submitted on 16 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

in older scholar texts¹— the different Syriac scripts are often called by names of Christian sects, which have used them in the past:

- *Nestorian*, which we call “East Syriac”;
- *Jacobite* (Syrian Monophysites), which we call “Serto” (ܣܪܬܘ);
- *Melchitic*, a variant of Estrangelo, not covered by this paper.

These names will not be used in this paper (native Syriac language speakers find them offensive).

Other scripts, with less problematic names, are “Estrangelo” (ܐܨܪܢܘܠܘܟܐ), *Peshito* (a variant of Serto) and *Mandean* (a very quadratic variant of Estrangelo, close to Arabic Kufic) the later two not covered by this paper.

1.2 Technicalities

Syriac presents the same typesetting problems as Arabic²: right-to-left writing direction, contextual analysis, multitude of vowels and diacritics. It should be noted that Syriac needs almost the double amount of vowels/diacritics than Arabic: in fact, Syriac uses many vowel systems, sometimes mixed, and Garshuni Syriac (Arabic written in Syriac script) uses all Arabic vowels and diacritics.

1.3 Solutions: T_EX, Ω

The ܣܒܪܐ (*Sabra* = “hope”, in Syriac) system is build upon T_EX and hence heritates all of T_EX’s power and flexibility. Nevertheless, the many problems involved in Syriac typesetting make the use of supplementary tools unavoidable: a pre-processor is used for ‘input encoding → font encoding’ translation, and contextual analysis; a multiple font loading mechanism is used to access all the signs, and TeX--XeT is necessary for switching writing directions.

This rather cumbersome way of adapting T_EX to Syriac will be obsolete once the Ω extension of T_EX³ is fully operational. Ω allows the use of 16-bit fonts (up to 65,536 characters) which is more than enough for Syriac; the contextual analysis and remapping of codes is handled by internal Lex-like filters and TeX--XeT is integrated by default.

With respect to the forthcoming availability of this new T_EX system, we will not enter too much into T_EX- and Metafont-specific techniques which become obsolete under Ω.

1.4 The Serto isolated *olaph*

The Serto letter *olaph*, when in isolated contextual form, can take two different shapes: “straight” ܐ and “curved” ܐ̣. Here are the rules for choosing between straight and curved form:

1. At word begin, the **curved** form is used;
2. Inside the word, or at word end, the **straight** form is used, except in the following cases:
3. When the letter *olaph* is preceeded by
 - (a) a *dolath* ܐܘܐ, or
 - (b) a *waw* ܘܐ, or
 - (c) a *waw* followed by a *dolath* ܘܐܘܐ, or
 - (d) a double *dolath*⁴ ܐܘܐܘܐ.

¹And even in more recent texts, like [8] (1980), still using only the terms ‘Jacobite’ and ‘Nestorian’.

²See [6] for a detailed discussion on the problems of Arabic typesetting.

³See [7] for a detailed discussion of Ω’s features, and follow the Ω project pointer in the <http://www.ens.fr> homepage to find out about the latest status of this very ambitious project.

⁴In a private communication, G. Kiraz mentions only the three first rules. The fourth one has been detected by the author in [4, p. 273].

Here are some exceptions to these rules, found in [3]:

- the word ܐܝܪ (ܐܝܪ = air), and its plural form ܐܝܪܝܢ, contradict rule 2: the curved form of *olaph* is used inside the word;
- the words ܘܠܘܦ (wolf), ܘܠܘܦܝܢ (combat) contradict rule 3-a: although *olaph* is preceded by a *dolath* it is typeset in straight form;
- the word ܘܠܘ (letter *waw*) contradicts rule 3-b: although *olaph* is preceded by a *waw* it is typeset in straight form.

It seems that all Syriac authors/typesetters do not follow the same rules: [2] uses only the curved form of *olaph*; the text given as example in [9, p. 181] uses a curved *olaph* at word begin and a straight one otherwise, EVEN when preceded by a *waw* or a *dolath*. In another document [5], also printed at the Imprimerie Nationale in Paris, one finds only straight *olaphs*, even at word begin...

According to f. H. Soumi⁵, the rules are by far more complicated than those given above, and a good knowledge of Syriac language is necessary to determine the form of *olaph* to be used. The author would be grateful for any supplementary information.

The input methods for the different forms of *olaph* are described in section 2.4, p. 5.

1.5 The Serto initial *olaph-lomad* ligature

A similar phenomenon as for the isolated *olaph* letter, occurs for the initial *olaph-lomad* ligature: the use of this ligature obeys to the same rules as the typesetting of curved *olaph* letter: this ligature is used:

1. At word begin, except in the following cases:
 - (a) a *dolath* ܐ, or
 - (b) a *waw* ܘ, or
 - (c) a *waw-dolath* pair ܘܐ, or
 - (d) a double *dolath*⁶ ܐܐ.
3. When the letters *olaph-lomad* are followed by *olaph*: in that case the first *olaph* is isolated curved and the letters *lomad-olaph* form a ligature. Example: ܐܘܠܘܢܐܝܪ (to bewail).

This rule seems to be even less followed as the one for the form of letter *olaph*; here are some exceptions, found in [3]: ܐܘܠܘܢܐ (letter *olaph*), ܐܘܠܘܢܐܝܪ (thousand) with the variant form ܐܘܠܘܢܐܝܪ.

The input methods for the different forms of *olaph* are described in section 2.4, p. 5.

2 Using Sabra

2.1 Requirements

To typeset in Syriac using ܐܘܠܘܢܐ, one needs a decent T_EX system (in this context, this would be a T_EX implementation featuring Peter Breitenlohner's TeX--XeT and an operational Metafont implementation), a relatively powerful machine (being able to run BigT_EX) and the ܐܘܠܘܢܐ (Sabra) package⁷, consisting of a preprocessor (*sabra*), fonts written in Metafont, and T_EX macros. The

⁵Personal communication.

⁶In the same private communication, G. Kiraz mentions only the three first rules. The fourth one has been introduced in analogy with the case of the isolated *olaph* letter.

⁷ܐܘܠܘܢܐ it is part of the long awaited version 1 of the ScholarT_EX package, together with *new* Greek, Arabic, Hebrew, Coptic and Akkadian cuneiform systems.

preprocessor is written entirely in gnu Flex, without using any system-dependent subroutines; hence it can be compiled in a straightforward manner on any platform having a gnu Flex executable and an ANSI C compiler.

Once **Ω** installed, typesetting is done in two steps: an input file is prepared using the syntax we will describe below; the preprocessor then reads this file, and produces a L^AT_ΕX 2_ε (or plain T_ΕX) file which then can be run through T_ΕX, in the usual way. Once Ω will be operational, there will be no pre-processor needed anymore.

2.2 Preparing the input file

If you wish to write your file in L^AT_ΕX 2_ε, you have to include the line

```
\usepackage{sabra}
```

in the preamble.

A **Ω** input file contains text, T_ΕX/L^AT_ΕX macros, and *preprocessor directives*. The latter concern only Syriac script. To type Syriac text you need to enter *Syriac mode*; this is done by the preprocessor directives <E> (Estrangelo), <S> (Serto) or <C> (East Syriac). To leave one of these modes, one enters </E> or </S> or </C>. ⁸

Once you are inside Syriac mode, you type Syriac text in Latin transcription, from left-to-right. No special indication needs to be given to T_ΕX about font or writing direction switching, this is done automatically by T_ΕX. ⁹ Section 2.4 describe the transcription you have to use as well as all other features of the preprocessor. ¹⁰

2.3 A text in a text in a text: nesting modes

Often one needs Latin, Greek, or other left→right insertions inside Syriac text; these insertions may again contain Syriac insertions, and so forth. If it weren't for the writing direction, it would be equivalent of switching from one script to the other. Unfortunately, the situation is more complicated: every writing direction is an environment, and paragraph breaking into lines depends on the rules of this environment. The only way to deal properly with such environment is to *nest* them: if you want to switch from Syriac to Latin, you have to decide:

1. if you are finished with Syriac and you are “returning” to Latin, in which case you leave Syriac mode by typing </E> (Estrangelo), </S> (Serto) or </C> (East Syriac);
2. if you are going to type a Latin insertion: in that case you enter (nested) Latin mode by typing <L>. To leave this mode, use </L>; this will bring you back to the script you were using before (Serto, Estrangelo or East Syriac).

Inside nested Latin mode the preprocessor is totally inactive: nothing is changed, up to the next </L>, <S>, <E>, or <C>. Similarly, when you are in (nested) Latin mode and enter into a (doubly nested) Syriac mode, everything is as in usual Syriac mode, with the only difference that when you leave it you are back in (nested) Latin mode, and so on. . . ¹¹

The figure below illustrates the nesting property of nesting modes: ¹²

⁸A few other similar directives concern other languages of scholarly interest: <A> for Arabic Naskhi, <H> for Biblical Hebrew, <K> for Akkadian cuneiform. Other writing systems provided by ScholarT_ΕX do not require a preprocessor.

⁹Nevertheless, when your text takes more than one paragraphs, you have to write the commands `\hfill` at the end of the first paragraph, and `\beginR` at the begin of the next one; this is an unfortunate shortcoming of TeX--XeT, which will be fixed in forthcoming versions of Ω.

¹⁰Nevertheless, the document preparation syntax will remain the same so that documents prepared for the current version of **Ω** will still be processable by Ω and produce exactly the same results.

¹¹The number of times you can enter a mode inside a mode, is given by the preprocessor internal variable `MAX_STATE_STACK_SIZE` which by default is set to 16. Besides switching from Syriac to Latin and back, you can also switch to Garshuni Syriac (see 2.6, p. 6) or from one Syriac script to another.

¹²For people with L^AT_ΕX experience, it may be enough to say that every **Ω** mode corresponds to a L^AT_ΕX environment.


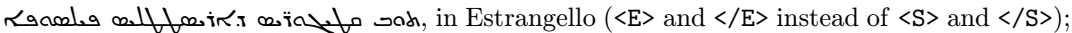
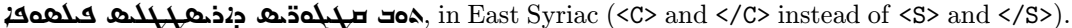
			...			
	<E>	</E>		
	<L>	</L>	
<S>	</S>

2.4 Transcription of Syriac consonants and long vowels

The Syriac transcription of letters (consonants and long vowels) is given in tables 1 (Serto), 2 (Estrangelo) and 3 (East Syriac), pages 9–10 of the appendix.

Here is a simple example of code producing non-vowelized Syriac text:

`<S>tuub qTygwRys d'rysTTlys pylsuup'</S>` will produce

- , in Serto;
- , in Estrangelo (<E> and </E> instead of <S> and </S>);
- , in East Syriac (<C> and </C> instead of <S> and </S>).

Some notes concerning the transcription of letters of tables 1–3:

There is no distinction between isolated, initial, medial and final forms; these are automatically applied by \TeX .

Some letters have multiple transcriptions: ' or aa for *olaph*, w or uu for *waw*, h or ee for *he*, y or ii for *yud*. These produce exactly the same results: choosing between them can make transcribed input more readable, especially when the text is not vowelized (for example, when used as *mater lectionis*, long vowels are more readable when transcribed as vowels, etc.).

In section 1.4, p. 2 we mentioned the two forms of Serto isolated *olaph*. If the rules described in that section are to be followed, then no extra precaution need to be taken: the transcription ' will produce a straight or a curved *olaph* depending on the context. To disobey the rules, two extra transcriptions are given: '| will produce a straight isolated *olaph*, and '@ a curved one, under ALL circumstances. So, for example, the exceptions seen in 1.4, p. 2 ܐܘܘܐ and ܐܘܘܐ are input as `<S>'@A'@ar</S>` and `<S>di'|bA1</S>` respectively.

In section 1.5, p. 3 we gaved rules about the typesetting of the *olaph-lomad* ligature. Once again, these rules are automatically applied by \TeX . To transgress them, one can use the same transcriptions as in the previous paragraph: '@ for a curved *olaph* (the logical choice, since we are necessarily at word begin) or '| for a straight one. Here are some example: `<S>'a1AhA'</S>` to obtain ܐܘܘܐܐܘܐ (god), `<S>'e1A'</S>` for ܐܘܘܐܐ (to bewail) but `<S>'@A1ep</S>` for ܐܘܘܐܐܐ (thousand).

Transcription s| stands for the two-form *semkath*. This letter will be typeset in isolated or final forms only, while usual s will have all four forms (isolated, initial, medial and final). The transcription s+ is used for the (two-form) letter *semkath* with tail.

2.5 Transcription of Syriac vowels and diacritics

There are many systems of Syriac vowelization, and often more than one of them are used simultaneously. \TeX provides four types of vowels:

1. Greek vowels (see table 4, p. 11);
2. Vowel points (see table 5, p. 11);
3. East Syriac vowels (see table 6, p. 11);
4. Jacob of Edessa vowels (these are not diacritics but real letters) (see table 7, p. 12).

These vowels can be used in the same context: a word can very well contain for example Greek and East Syriac vowels: ܢܝܘܒܐ ܚܘܩܠܐ ܘܚܘܩܠܐ ܘܚܘܩܠܐ (listen the words of their king).

In two cases (inverted Greek vowels and Jacob of Edessa vowels), there are alternate input methods. If one has to typeset texts which have more inverted Greek vowels than regular ones,

or more Jacob of Edessa vowels than Greek ones, one can enter into “inverted vowel” (or “Jacob of Edessa vowel”) mode. This is done by typing `<INVVOW>` (resp. `<JACVOW>`). Once entered in one of these modes, the transcription of vowels changes, as indicated in the table. This behaviour of the pre-processor continues until the user exits the mode (by typing `</INVVOW>` or `</JACVOW>` respectively).

2.6 Transcription of Garshuni letters and diacritics

Garshuni is Arabic written in the Syriac script. Since the Arabic language has a few more sounds than Syriac, new letters had to be introduced: these are pre-existing Syriac letters with additional dots or strokes¹³. The Garshuni transcription used in **ܐܡܠܐ** is exactly the same as in the 3l1mÖ0A10A (*Al-amal* = “hope”, in Arabic) system, the Arabic part of ScholarTeX system.

This means that the transcription used to typeset Arabic *language* in Syriac script, is DIFFERENT from that used to typeset Syriac *language*, although the letters themselves are mostly the same. The transcription used for Garshuni is described in tables 11, p. 14 (Serto), 12, p. 15 (Estrangello) and 13, p. 16 (East Syriac), for the consonants and long vowels and in table 10, p. 13 for the Arabic vowels and diacritics.

This transcription is only valid in “Garshuni mode”: to enter Garshuni mode, one has to do the following steps: (a) entering into Syriac mode (by typing `<S>` for Serto, `<E>` for Estrangello or `<C>` for East Syriac), and (b) enter into Garshuni mode, by typing `<G>`. So, for example, the famous Arabic “welcome”

ܐܘܠܐܝܢܐ ܘܐܘܠܐܝܢܐ

is transcribed `<A>'aahlAaN wa sahlAaN` in 3l1mÖ0A10A; to typeset it in Syriac Serto script we just have to write `<S><G>'aahlAaN wa sahlAaN</G></S>`. Here is the result:

ܐܘܠܐܝܢܐ ܘܐܘܠܐܝܢܐ

The reader must realize that these “modes” have to be *nested*: if one enters Syriac mode and then Garshuni mode, then one must *first* leave the later and *then* the former.¹⁴

2.7 Stretching letter connections (0h"diS1k)

As in Arabic, Syriac words are never hyphenated. To balance white space on each line, one stretches connections between initial and medial, or medial and final forms of letters. This stretching is called 0h"diS1k “keshideh” (a Persian word, derived from the verb 0ndiS1k, “to stretch”, see [1, p. 171]).

In **ܐܡܠܐ**, stretching of letter connections is not activated by default. To activate it, one uses the preprocessor directive `<KESHIDEH>` (and `</KESHIDEH>` to deactivate it). This is not a “mode” in the sense of `<G>` (Garshuni mode) or `<L>` (Latin mode) and hence doesn’t need to be nested with respect to Syriac mode.

Here is an example of the use of letter connection stretching:

¹³There seem to be many ways of adapting Syriac script to the Arabic language; the author has chosen to follow the one suggested by G. Kiraz. Have there been Persian or Turkish texts written in Syriac script? The author would be grateful on any information related to this question.

¹⁴This is not the case for the “inverted vowel” mode and the “Jacob of Edessa vowel” mode: these can be entered and exited at any time, without nesting with Syriac mode. To be more precise we should call these “pseudo-modes”, or “global flags”.

□

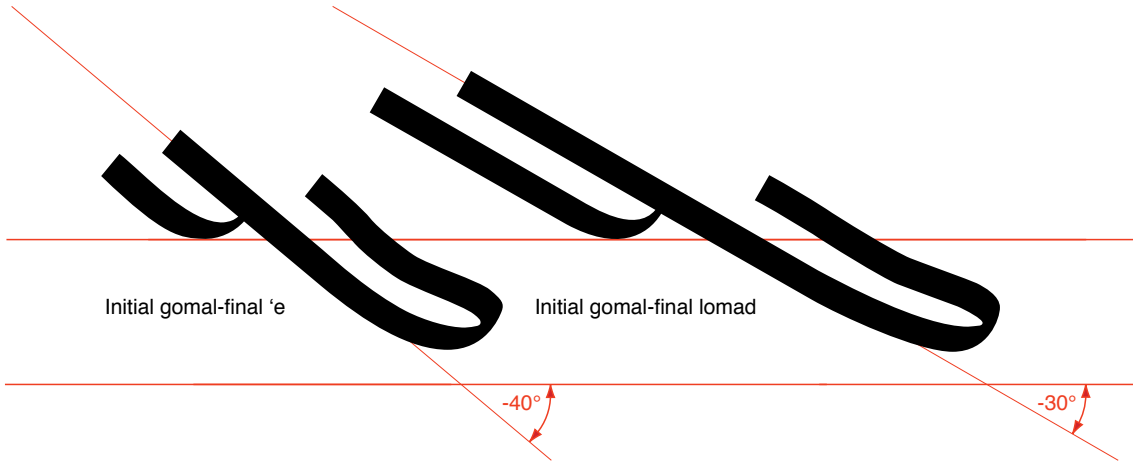


Figure 1: The Serto isolated *gomal-lomad* and *gomal-'e* ligatures

□

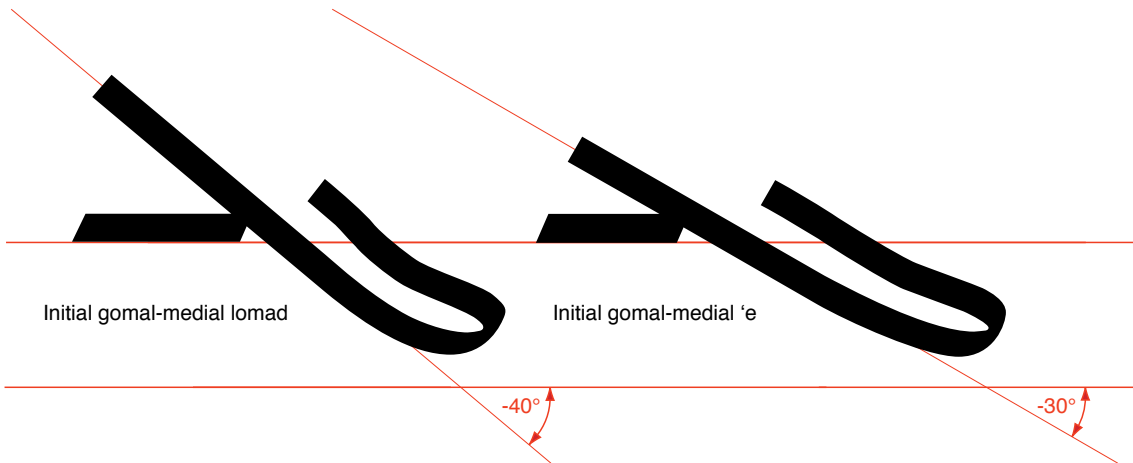



Figure 2: The Serto initial *gomal-lomad* and *gomal-'e* ligatures

What happens now when *gomal* is followed by an ‘e rather than by a *lomad*? Once again the *gomal* must have the same angle as this letter.

This means that we need two *gomals*: one for the (final) *lomad*, with an angle of 30°, and one for the (final) ‘e, with an angle of 40°. The amusing part is that the former will fit also exactly to the (medial) ‘e and the later to the (medial) *lomad*. These ligatures are illustrated on figures 1 (*gomal* followed by final letters) and 2 (*gomal* followed by medial letters).

Of course  chooses automatically the right form of letter *gomal*.

4 Tables

4.1 Syriac consonants and long vowels

'	b	g	d	D	h, ee	w, uu	z	x
T	y	k	l	m	n	s	s+	'
p	Y	q	r	R	S	t	'	'@

Table 1: Syriac Serto letters and their input codes

'	b	g	d	D	h, ee	w, uu	z	x
T	y	k	l	m	n	s	s+	'
p	Y	q	r	R	S	t	'	'@

Table 2: Syriac Estrangelo letters and their input codes

'	b	g	d	D	h, ee	w, uu	z	x
T	y	k	l	m	n	s	s+	'
p	Y	q	r	R	S	t	'	'@

Table 3: East Syriac letters and their input codes

In normal mode:	a	A	e	i	u
In inverted mode:	<INV>a	<INV>A	<INV>e	<INV>i	<INV>u
In normal mode:	<INV>a	<INV>A	<INV>e	<INV>i	<INV>u
In inverted mode:	a	A	e	i	u

Table 4: Greek vowels and their input codes (type <INNVOW> to enter “inverted mode”, </INNVOW> to exit)

<.>, .a, .o	<_.>, .e, .i

Table 5: Vowel points and their input codes

/a	/A	//A	/E
//E	e, /e	/o	/u

Table 6: East Syriac vowels and their input codes

In default mode:	<JAC>a	<JAC>E	<JAC>e	<JAC>I
In Jacob of Edessa mode:	a	E	e	I
In default mode:	<JAC>i	<JAC>U	<JAC>u	<JAC>u/
In Jacob of Edessa mode:	i	U	u	u/

Table 7: Jacob of Edessa vowels and their input codes (type <JACVOW> to enter “Jacob of Edessa vowels mode”, </JACVOW> to exit)

<PLU>, <SEYAME>	<_PLU>, <_SEYAME>	_, <_SHWAYA>	_, <SHWAYA>
<QUS>, <QUSHOYO>	<RUK>, <RUKAKHA>	<_LIN>	<^LIN>
	<BAREKH>	<QSO>	

Table 8: Miscellaneous Syriac diacritics

;	,	+:	\:	/:
:, : ¹⁵	.	::	<<	>>

Table 9: Syriac punctuation

4.2 Arabic written in Syriac script (Garshuni)

a	"a	i	u	:	*	aN	iN
uN	*a	*"a	*i	*u	*aN	*iN	*uN

Table 10: Garshuni vowels and their input codes (type <G> to enter “Garshuni mode”, </G> to exit)

¹⁵The transcription |: for : becomes necessary in Garshuni mode, because of a conflict with one of the *sukun* diacritic (see table 10).





































				
	A	'A	"A	'a
				
'i	'u	'y	b	"t, "h
				
t	t_	g	H	h_
				
d	d_	r	z	s
				
S	Y	D	T	Z
				
'	G	p, f	q	k
				
l	m	n	h	w, U
				
I, y				

Table 11: Garshuni Serto letters and their input codes (type <G> to enter “Garshuni mode”, </G> to exit)

l	A	'A	"A	'a
'i	'u	'y	b	"t, "h
t	t_	g	H	h_
d	d_	r	z	s
S	Y	D	T	Z
'	G	p, f	q	k
l	m	n	h	w, U
I, y				

Table 12: Garshuni Estrangello letters and their input codes (type <G> to enter “Garshuni mode”, </G> to exit)

l	A	'A	"A	'a
'i	'u	'y	b	"t, "h
t	t_	g	H	h_
d	d_	r	z	s
S	Y	D	T	Z
'	G	p, f	q	k
l	m	n	h	w, U
		I, y		

Table 13: Garshuni East Syriac letters and their input codes (type <G> to enter “Garshuni mode”, </G> to exit)

References

- [1] Daniel Berry and Johny Srouji. Arabic formatting with ditroff/ffortid. *Electronic Publishing—Origination, Dissemination and Design*, 5(4):163–208, 1992.
- [2] C. Brockelmann. *Syrische Grammatik, mit Paradigmen, Literatur, Chrestomathie, und Glossar*. VEB Verlag Enzyklopädie, Leipzig, 1938.
- [3] S.J. Costaz, L. *Syriac-English Dictionary*. Imprimerie Catholique, Beyrouth, 1963.
- [4] K. Georr. *Les catégories d’Aristote dans leurs versions syro-arabes*. Institut Français de Damas, Beyrouth, 1948.
- [5] A. Guillaumont. *Sémitismes dans les Logia de Jésus, retrouvés à Nag-Hamâdi*. Société Asiatique, Librairie Orientaliste Paul Geuthner, Paris, 1958.
- [6] Y. Haralambous. Typesetting the holy Qur’ân with T_EX. In *Proceedings of the 2nd International Conference on Multilingual Computing—Arabic and Latin script (Durham)*, 1992.
- [7] Y. Haralambous and J. Plaice. First applications of Ω : Greek, Arabic, Khmer, Poetica, ISO 10646/UNICODE, etc. In *Proceedings of the 15th T_EX Users Groups Annual Meeting (Santa Barbara)*, 1994.
- [8] A. Nakanishi. *Writing systems of the World*. Charles E. Tuttle Company, Tokyo, 1980.
- [9] Imprimerie Nationale. *Les caractères de l’Imprimerie Nationale*. Imprimerie Nationale, Éditions, Paris, France, 1990.