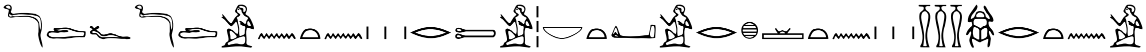
Egyptian Hieroglyphs in Unicode plain text

A note on a suggested approach

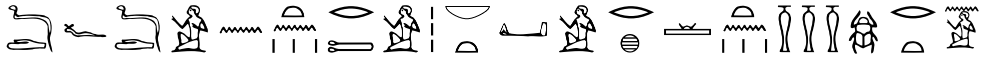
L2/15-069

Background

Basic Egyptian Hieroglyphs have been defined in Unicode since version 5.2 (2009) but there has been no agreed means of presenting hieroglyphs written in plain text that takes into account the arrangements of hieroglyphs into groups. This means a web browser or word processor will show plain text hieroglyphs as



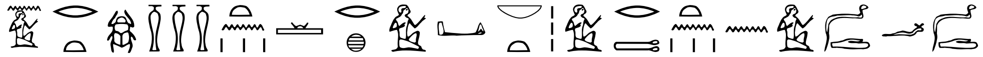
rather than an authentic rendering in Middle Egyptian style:



[This example is taken from the 'The Autobiography of Admiral Ahmose', from his early 18th Dynasty tomb.]

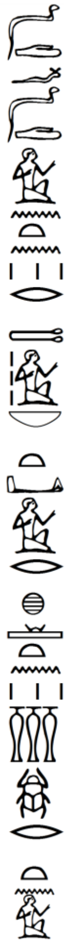
The use of groups/clusters is intrinsic to the hieroglyphic script so the first 'linear' writing is highly anachronistic. Groups are primarily an aesthetic characteristic of the writing of hieroglyphs, for the most part the linear form is perfectly readable and nothing is lost however odd it may look. Search engines can use the linear form, potentially enhanced by semantics associated with the various signs.

The Egyptian Hieroglyphic script can be written vertically or horizontally, in both cases in left to right and right to left directions. Symmetry pervades and the characters follow direction so the text above written right to left is a simple transformation:



In modern representations, left to right is the norm. In ancient sources right to left dominated, although both directions appear in surviving texts.

The example of vertical writing to the right illustrates instances where grouping of hieroglyphs can be slightly different to that of the horizontal script in some instances despite much commonality. Modern works in Egyptology often publish vertical writing in Middle Egyptian horizontal style.



Late Egyptian (notably during the Ramesside period) adopted some features of vertical writing to the horizontal script so there are differences from the earlier (and later) style.

This note primarily focusses on horizontal writing in the Middle Egyptian style. The suggestions here are supported by analysis of a word list containing c. 20,000 spellings of words in hieroglyphs, data that was not available when Egyptian Hieroglyphs were added to Unicode in 2009. The largest primary sources for the data are references from *A concise dictionary of Middle Egyptian, Egyptian Grammar*, and *Wörterbuch der Ägyptischen Sprache.*

Current practice in encoding hieroglyphic texts is largely based on the Manuel de Codage (MdC) system originally designed in the 1980s before the Unicode era for typesetting Egyptian Hieroglyphic texts. MdC works with a catalogue of hieroglyphs identifying encoded signs as alphanumeric codes A1, A2, … B1, B2, using an extension of the Gardiner sign list such as Hieroglyphica. Mnemonic equivalents e.g. 't' for X1 etc. based on phonetic values can also be used. See Unicode TN#32 for MdC character mappings. Some arrangements of signs are encoded as ligatures, for instance 𓆓&𓂧 meaning 𓆓𓂧 , but most groups are encoded as a stack of rows, each with one or more hieroglyphs. For instance A1*B1:Z2 is traditional MdC for the group 𓀀𓁐𓏪 - (𓀀*𓁐:𓏪 is a Unicode/MdC equivalent although Unicode notation is rarely seen to date).

Note: MdC has various other features to apply to hieroglyph typesetting including encodings to shade hieroglyphs and clusters, rotate signs, and mirror signs. Implementations of MdC may add extra features, e.g. JSesh adds a way to specify exactly where and at what scaling a hieroglyph participates in a group. For the most part these are not features to be used with plain text and are naturally part of one or other higher level protocol (although see below).

Encoding hieroglyphs in plain text

Hieroglyphs

Unicode currently identifies 1071 characters in the BASIC EGYPTIAN HIEROGLYPHS set.

There are also several hundred candidate characters from the Old Kingdom to the Ramesside period with references identified in my current dataset, with more existing in other sources but not yet included. The explosion in new hieroglyphs appearing during the Greco-Roman period (with over 10,000 catalogued) is another topic. A large majority of the potential additions consist of determinatives and ideograms with acceptable alternatives in the existing repertoire. An expanded repertoire has no known implications for the plain text encoding method suggested here so there is no linkage to take into account.

MdC encoding works with these larger catalogues so continues to be essential where hieroglyphs are not yet encoded in Unicode. A higher level protocol that combines Unicode with traditional MdC would be useful but as far as I know no such protocol is in common use yet.

Nevertheless for many practical purposes the limited Unicode repertoire is sufficient for Middle Egyptian and many Late Egyptian applications – even a large work such as the Topographical Bibliography (an ongoing century-old project) should be able to use Unicode Basic Hieroglyphs for most material.

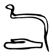
Within the basic encoded set, there is a small number of hieroglyphs with known variations in orientation or direction in ancient use. The most commonplace example is the 𓏛 character whose mode of use evolved around Dynasty 12 to appear in a vertical variation 𓏜 alongside continuing use of 𓏛. Both 𓏛 and 𓏜 were therefore encoded separately in the basic set as are a small number of other variants, exx. and , and , and . Not all variants are encoded, for instance 𓊤 (EGYPTIAN HIEROGLYPH P008, 'oar') has an unencoded variant ; 𓆠 (EGYPTIAN HIEROGLYPH K006, 'fish-scale') has an unencoded variant . MdC provides codes to mirror hieroglyphs and/or rotate through, 90°, 180°, or 270° to allow for these geometric variants of signs.

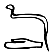
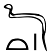
It is interesting to note that in sample data, the horizontal oar is only seen in a cluster context so one solution in this instance is to render MdC 𓐙:𓊤 as . However there are observed cases such as - mirrored 𓐙 (EGYPTIAN HIEROGLYPH AA011) – where this technique doesn't apply.

Plain text Unicode could mimic MdC and provide equivalent general purpose equivalents to the MdC MIRROR and ROTATE codes, noting these could have applications beyond Egyptian Hieroglyphs (this was proposed in N1647, Everson, 1997). However to avoid this complication, the suggestion here is to investigate the small number of instances where these geometric variants occur consistently in the corpus and introduce variation selectors for these if and only if context is not sufficient. Any quirks associated with specific scribal hands or artistic license can properly be left to higher level protocols.

MdC also defines four pseudo-signs in different sizes , , and to be used when a hieroglyph is unreadable, e.g. . Shading can also be applied to individual hieroglyphs to indicate damage, e.g. and . A further shading effect can be applied to quadrants of a sign or cluster e.g. . These editorial devices are largely a matter for higher level protocols and not applicable to this plain text suggestion. However the four pseudo-sign characters could be useful in a wider Unicode context for all scripts to deal with missing characters and plain text hieroglyphs could benefit from such a mechanism should one be defined.

Ligatures

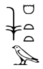
MdC '&'-type ligatures such as  (combining and) are essential for a plain text mechanism. Some other examples are , , , , , and .

Where an MdC '&'-type ligature is explicitly required we need a LIGATOR character. In what follows I'll use '^' to represent this hypothetical character so ^ means  . These ligatures can involve more than two hieroglyphs, for instance ^ ^ means  .

For Middle Egyptian rendering, at present about 70 ligatures are identified as essential in my large data set.

I suggest introducing a new control character EGYPTIAN HIEROGLYPHIC SIGN LIGATOR for LIGATOR implementation. This character is explicitly defined to form ligatures of two or more characters.

Experimental work has used ZERO WIDTH JOINER to implement LIGATOR. This is currently the only practical option for Unicode implementations although unless I've missed something, ZWJ appears to be intended to only apply to two character ligatures. As far as I know, use of ZWJ has not escaped into the wild yet for hieroglyph ligatures in fonts but it's so obvious for important ligatures like that I'm surprised it hasn't happened yet. In short, clarification is fairly urgent.

Late Egyptian sees some stylistic changes in writing hieroglyphs horizontally that borrow from conventions in vertical writing, there are clusters with more rows and signs in addition to the familiar Middle Egyptian elements. MdC allows for sub-clusters (in parenthesis) so one can write M23*(N41:X1:X1):G37 to obtain a cluster  (this example is from Amarna stela U, 18th Dynasty) – an arrangement that isn't obtainable in the simple MdC row-based cluster mechanism that works well for the Middle Egyptian horizontal writing style. Rather than require sub-clusters for plain text, LIGATOR can be used here (N41^X1^X1).

Analysis of Middle Egyptian word list data from horizontal style hieroglyphs and hieratic transcription shows very few instances of sub-clusters, all readily addressed using LIGATOR.

I therefore suggest that LIGATOR is used in preference to introducing the complication of an MdC sub-cluster mechanism into Unicode. This approach can be reviewed if necessary when there is a sufficient corpus of machine-readable Late Egyptian and vertical script source material available for analysis.

Plain text punctuation

Word spacing, end of sentence markers, and other forms of punctuation were not used in ancient writing of hieroglyphs, although determinatives and other constructs make breaks apparent to the trained eye. Nevertheless it is useful to use word-breaking characters in plain text Unicode so the suggestion is to use ZERO WIDTH SPACE to separate words where possible. This helps with text analysis, search and presentation. In applications such as modern grammars and dictionaries, visible spaces are useful. Guidance on plain text writing should deal with various General Punctuation characters.

MdC does have a code for word break although this is often not used in encodings of texts seen to date.

All hieroglyph groups can be thought of as ligatures in the Unicode sense, not only the MdC 𓆓𓂧 '&' type ligatures but likewise the row-formatted groups like 𓀀𓁐𓏥 . It is therefore useful in plain text to insert a character between two hieroglyphs when it's necessary to indicate a default ligature must not be formed. The obvious candidate character for SEPARATOR is ZWNJ so 𓀀[ZWNJ]𓁐𓏥 can be used to assert that 𓀀 𓁐𓏥 or similar is required. In practical situations, this non-joiner is not required all that often so the plain text doesn't get cluttered.

The cartouche and other enclosure characters act as punctuation like parentheses and imply a cluster break. They imply visual enclosure of the text inside, for instance 𓍹𓇳𓏠𓆣𓍺 implies rendering as 𓍹𓇳𓏠𓆣𓍺 . It is desirable that plain text presents the full enclosure but acceptable that this is an optional feature for a given font. OpenType can be use to implement such as scheme as open and close are explicitly marked. Similar remarks apply to the Serekh and Hwt enclosures.

Implied clusters

A run of hieroglyphs such as 𓂋𓊃𓀀𓏪 has a natural clustering/group structure 𓂋𓊃𓀀𓏪 , there is no real need to be explicit about the cluster by inserting control codes into plain text. Having said that, there are over 5000 different cluster groups of hieroglyphs in the total test data, including many instances of the same run of hieroglyphs being written as different groups. Many of these are rare, conventional patterns dominate, and it is possible to make some simplification of this situation. If, for instance, we limit attention to groups used in Gardiners 'Egyptian Grammar', it is possible to define a set of rules which associate sequences of hieroglyphs to their 'natural' group/cluster layouts which match the presentation style used throughout the book – 'Gardiner Typography'. If the intent is to display a run of hieroglyphs in the Gardiner style, most of the time there is no need for MdC-like control characters, the groups are implicit in the sequence. This 'Gardiner Typography' can be implemented in an OpenType font using GSUB, or in specialist software.

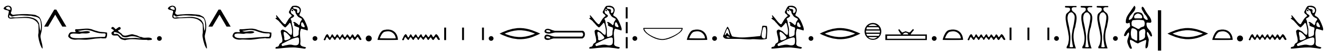
To understand why this approach is successful, it's important to be aware that clusters are not written at random in the hieroglyphic script, conventions abound and persist despite underlying linguistic changes over the ancient 3000 year period when hieroglyphs were in active use.

The style of *Egyptian Grammar* hieroglyphic writing is commonplace for modern representations of Middle Egyptian, indeed the Champollion Grammaire and Dictionnaire from the 1830s mostly use very similar clustering conventions. Therefore many additional sources can be used to refine and expand the scheme somewhat (I've sometimes referred to this as 'Oxford Typography' as it follows the Gardiner style and uses more source material using the Oxford font as well as other references). A variation better suited to Late Egyptian material can be imagined – 'Ramesside Typography'.

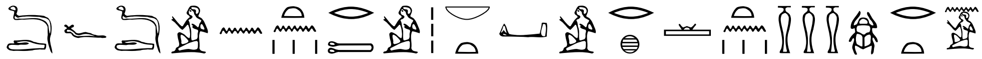
The point here is so long as a typography is well defined and documented, it is reasonable and useful to use runs of hieroglyphs without need for positional control characters. Punctuation as discussed above gives limited control over clustering. The detail of what constitutes a useful typography in this sense is a matter for consensus for the user base.

I coined the term '**Simplified Egyptian**' several years ago to describe a plain text encoding system entirely based on ligatures and implied clusters. The benefit of such as system is ease of use and its broad domain of applicability. Simplified Egyptian is easy to compose in text editors, word processors and other applications without the need for specialist input methods. Most of 'Egyptian Grammar' and a digital version of a major work such as Topographical Bibliography could be encoded in this system for online or other publication. It is also useful when defining higher level protocols for specialist software.

With codes exposed, a sequence of characters in Simplified Egyptian looks something like



where '.' is used to visualize word separator, │ for SEPARATOR and ^ for LIGATOR (relative to a draft 'Gardiner typography'). To compose this sequence it was sufficient to add 2 ligatures plus a SEPARATOR to choose 𓆣𓂋𓏏 in preference to the 𓆣𓂋𓏏 default implied clustering relative to the draft typography. As before, this sequence renders as



A fully documented typography can be applied to fonts and other software that implements custom rendering so the clustering behaves exactly the same way but uses whatever stylistic features a font may provide. This enables a diversity of compatible hieroglyphic font families. A choice of fonts based on highly detailed hieroglyphs, plain functional hieroglyphs, 'filled' hieroglyphs, or color hieroglyphs would enrich applications of the script.

There are already instances where GSUB has been used in hieroglyphic fonts on an ad-hoc basis so it is desirable to have a well-defined model before hack solutions proliferate.

Explicit clusters

Implied clusters in any given typography cannot entirely mirror MdC. It also seems to be a really bad idea to proliferate typographies if special clusters are essential for specific texts. When encoding hieroglyphic writings into plain text, it can be very useful to be assertive about a group/cluster layout, not rely on how a typography/font interprets a sequence however well-defined and deterministic the method.

It is therefore essential to allow an explicit MdC type group/cluster specification in plain text so Unicode is not regarded as a backwards step by Egyptologists. MdC A1*B1:Z2 could be written using Unicode characters - 𓀀*𓁐:𓏥 but this abuse of '*' and ':' is unacceptable in Unicode terms so two new control characters are required:

Name	MdC
EGYPTIAN HIEROGLYPHIC SIGN JUXTAPOSITIONER	*
EGYPTIAN HIEROGLYPHIC SIGN SUBORDINATOR	:

Plain text encoding with explicit clusters allows MdC encoded material to be directly re-used in Unicode. This system fulfils a different function to Simplified Egyptian although the two share LIGATOR and SEPARATOR and punctuation considerations. This explicit system could be called UMdC (Unicode MdC).

UMdC could adopt the convention that a SEPARATOR should be used to separate all clusters but in general the cluster boundaries are implicit so this is not needed. MdC ligatures (using LIGATOR) can participate in the arrangement exactly like individual hieroglyphs.

Fonts devised for Egyptologists could allow formation of arbitrary clusters based on UMdC using a combination of OpenType lookups, although for a given typography model a subset will often be acceptable.

It should be noted that, as is the case now with MdC, editing of UMdC texts will likely often be done with the aid of specialist software rather than requiring direct editing of the plain text elements. The user interface of such software can simplify transcription of sources into machine readable text (e.g. selecting ligatures and clusters from palettes and linking to phonetic elements of the language).

Summary

The discussion above amounts to the following items to be addressed for Unicode.

1. Identify essential geometric variants of basic hieroglyphs and decide if contextual processing is sufficient or whether a variation selector should be used.
2. Allocate a LIGATOR character to join two or more hieroglyphs
3. Allocate a (cluster) SEPARATOR character used to separate clusters inside a word/text unit (Is ZWNJ acceptable or is there some reason to allocate a new character?)
4. Summarize punctuation guidelines (including SEPARATOR, ZERO WIDTH SPACE, other space and general punctuation characters).
5. Add new characters EGYPTIAN HIEROGLYPHIC SIGN JUXTAPOSITIONER and EGYPTIAN HIEROGLYPHIC SIGN SUBORDINATOR to enable MdC cluster constructs to be included in plain text.

Simplified Egyptian addresses casual and much professional use of Ancient Egyptian in plain text. UMdC gives a rigorous way of defining clusters in plain text to the same level as established MdC practices.

Sufficient data exists in MdC format to define sample typographies and produce copious examples of how this system works so there is every prospect of gaining consensus among specialists once there exist working implementations with clear documentation.

Acknowledgements

Conversations with Jaromir Malek and Vincent Razanajao (former and current editor of the Topographical Bibliography) have been useful in clarifying the scope and applicability of this suggestion.

Feedback from Andrew Glass (Microsoft) has been very helpful in preparing this note.

Selected references

Allen, James P. 1999. *Middle Egyptian: an introduction to the language and culture of Hieroglyphs.* Cambridge: Cambridge University Press. ISBN 0-521-77483-7 [**ME**]

Collins, Lee. 2009. Unicode TN#32 Unicode Technical Note #32 MAPPING BETWEEN MANUEL DE CODAGE AND UNICODE EGYPTIAN HIEROGLYPHS/ http://www.unicode.org/notes/tn32/

Erman, Adolf (editor), and Herman Grapow (editor), 1971 (1926-1963). *Wörterbuch der Ägyptischen Sprache*. Berlin: Akademie Verlag. [**WB**]

Everson, Michael. 1997-08-25. *Encoding Egyptian Hieroglyphs in ISO/IEC 10646-2.* N1636 http://www.dkuug.dk/JTC1/SC2/WG2/docs/n1636/n1636.htm

Everson, Michael and Bob Richmond. 2007-04-10. *Proposal to encode Egyptian Hieroglyphs in the SMP of the UCS.* http://www.unicode.org/L2/L2007/07097-n3237-egyptian.pdf

Faulkner, Raymond O. 1986 (1962). *A concise dictionary of Middle Egyptian.* Oxford: Griffith Institute. ISBN 0-900416-32-7. [**DME**]

Gardiner, Alan H. 1957. *Egyptian Grammar: being an introduction to the study of hieroglyphs.* 3rd edition. London: Oxford University Press. [**EG**]

van den Berg, Hans. 1997. *"Manuel de Codage": A standard system for the computer-encoding of Egyptian transliteration and hieroglyphic texts.* [Leiden]: Centre for Computer-Aided Egyptological Research. http://www.catchpenny.org/codage/