

Observations about L2/16-090

BOB RICHMOND, Version 2016/05/03

L2/16-090 contains three parts:

1. Comments on three control characters for Egyptian Hieroglyphs 2016-04-18 (Mark-Jan Nederhof and Vinodh Rajan) – **NR**.
2. Notes about the Proposal to encode three control characters for Egyptian Hieroglyphs (Stéphane Polis and Serge Rosmorduc) – **PR**.
3. Remark on the ‘Proposal to encode three control characters for Egyptian Hieroglyphs’ ... 2016-04-18 (Tonio Sebastian Richter, Ingelore Hafemann, Simon Schweitzer) – **RHS**.

Introduction

All comments in L2/16-090 (25th April 2016) relate to L2/16-018R, a clarification of the L2/15-123 **Proposal to encode three control characters for Egyptian Hieroglyphs** (4th May 2015) in which the three control characters were introduced. These are the only comments received by UTC so far since the proposal was published a year ago in Spring 2015.

I have not had time to review all points made in L2/16-090.

I began sending detailed discussion notes to contributors to L2/16-090 a week ago to help clarify some points and potential issues but have not yet received feedback about the first notes from most as of writing. I hope to update this observations document after feedback is received.

Control characters for Plain Text

All three responses appear comfortable with a control character approach in principle.

PR and **RHS** express support for EGYPTIAN HIEROGLYPH HORIZONTAL JOINER and EGYPTIAN HIEROGLYPH VERTICAL JOINER. **NR** “have no great objection” to these but recommend they are not included in Unicode until research is done to develop a new way of encoding spatial arrangements of hieroglyphs for Unicode to extend the scope of plain text beyond L2/16-018R.

None of the responses claim that a more elaborate treatment of plain text could not be addressed by adding to L2/16-018R in the future. If a need was established. Nor is there any suggestion that extending the hieroglyph repertoire might change anything.

NR, **PR** and **RHS** all express reservations about EGYPTIAN HIEROGLYPH LIGATURE JOINER. I’ll attempt to treat some issues raised below.

One point in **NR** is easily cleared up. Monograms such as  (D059) have been treated as separate characters in traditional printing (e.g. Oxford font as used by Gardiner and Topographical Bibliography) and digital encodings (MdC, Hieroglyphica, Unicode 2009). There was no suggestion in L2/16-018R to change this traditional practice. The **NR** comments on superposition (page 2 etc.) are therefore irrelevant to the discussion. Nevertheless, monograms could be discussed as part of any proposal to extend the hieroglyph repertoire if there is interest in using a new joiner to create monograms instead of encoding them separately.

NR assert “it is essential that we do not confuse the order in which the signs are written”. **PR** raise the point that ,  and  can all have the same reading in different situations (in transliteration aq wnm). In L2/16-018R these would be written  + ,  +  and  so the second case would see the two hieroglyphs in reverse order in the character sequence. **NR** point out the well-known fact that  ( + ) is sometime used instead of  ( + ) although it can be read as  before  (the ambiguity to be resolved by the reader).

For a human reader of these elements in Unicode plain text hieroglyphic L2/16-018R there is no obvious problem in these examples, the reader simply needs to know relevant characteristics of the writing system in order to read it. Likewise, software that processes this text and interprets it needs to apply rules. Unicode encodes the writing elements, not their meaning.

If, after noting these points, there remains some problem relating to the **order of hieroglyphs** in L2/16-018R as suggested there might be in L2/16-090, the problem needs to be understood.

Unicode Plain Text and traditional encodings such as MdC

It is important to understand the difference between a Unicode plain text hieroglyphic writing system and an encoding system such as MdC. An MdC application typically uses sequences of hieroglyph codes and typesetting codes to render a graphic showing the hieroglyphic text using its own cluster layout rules. Unicode plain text implements a writing system, usually rendered using OpenType fonts that contain their own rules determining how rendering takes place.

This means the L2/16-018R control characters are used in a different context to their ancestors in MdC and it is a mistake to treat the two contexts as interchangeable. The term JOINER was chosen to help emphasise

this difference. Consider MdC ns:niwt which JSesh software renders as  by placing one sign from its font entirely above the other. A Unicode font using EGYPTIAN HIEROGLYPH VERTICAL JOINER in plain text might kern and render as  or choose to follow JSesh. A choice for the font designer. The question of what makes for good hieroglyphic fonts for specific purposes is then down to the user community, not a matter for encoding. In this specific instance an MdC application has no option but to use the ‘&’ control to provide a cluster  whereas in Unicode it is not necessary to use EGYPTIAN HIEROGLYPH LIGATURE JOINER unless it is decided that both the JSesh form and the kerned form both need to be part of the writing system.

L2/16-018R characters are used to form clusters from sequences and should not be thought of as typographic operators in the MdC sense. Thus  is encoded as the sequence  *  :  and  is encoded  +  +  + . In neither case is there some implied arithmetical calculation of relative proportion of the components as in MdC implementations, the font itself provides its own ‘take’ on how the sequence is rendered. Different fonts may choose to render in different ways according to their purpose but all should follow guidance on the meaning of defined sequences. This guidance can be evolved within the Egyptology community.

NR discusses issues with MdC use of ‘&’ then transfers the criticism to EGYPTIAN HIEROGLYPH LIGATURE JOINER wholesale without considering the different context, capabilities and requirements of a plain text system. This makes it difficult to tease out any actual potential problems they may have identified.

It would be useful to understand what the contributors to L2/16-090 see as problems with L2/16-018R once the different context of use to MdC is taken into account.

Limits of plain text

A plain text hieroglyphic writing system absolutely cannot meet the rich requirements of the scholarly community. It is impossible to properly publish many inscriptions and paintings without sophisticated higher level protocols. Just as with other writing systems and more so given the 2 dimensional nature of many ancient sources.

So some practical technical questions are:

- Does L2/16-018R provide a sufficiently rich writing system to allow for an acceptably wide range of applications?
- Does L2/16-018R provide an adequate plain text substrate for the construction of higher level protocols suitable for advanced applications?
- Does L2/16-018R provide the basis for development of a more advanced plain text system if a future need for additions is proven for advanced applications?
- What guidance for implementation is required?

And scope of applicability:

- What potential problems with L2/16-018R if any are identified for 'modern' uses of hieroglyphic writing such as grammars, dictionaries and learning where there is no need to exactly reproduce an ancient document?
- What potential problems with L2/16-018R if any are identified for transliteration of hieratic sources into hieroglyphic?
- Is it possible to see some actual data or analysis of the TLA and Ramses databases so the scale and character of the perceived potential problems can be assessed? At present this crucial information is undisclosed.

These are key issues not addressed in the L2/16-090 document.