

**Draft** Unicode® Standard Annex #57

# UNICODE EGYPTIAN HIEROGLYPH DATABASE (UNIKEMET)

Version	Unicode 16.0.0
Editors	Michel Suignard
Date	2024-02-06
This Version	<a href="https://www.unicode.org/reports/tr57/tr57-2.html">https://www.unicode.org/reports/tr57/tr57-2.html</a>
Previous Version	<a href="https://www.unicode.org/reports/tr57/tr57-1.html">https://www.unicode.org/reports/tr57/tr57-1.html</a>
Latest Version	<a href="https://www.unicode.org/reports/tr57/">https://www.unicode.org/reports/tr57/</a>
Latest Proposed Update	<a href="https://www.unicode.org/reports/tr57/proposed.html">https://www.unicode.org/reports/tr57/proposed.html</a>
Revision	2

**Summary**

*This document describes the organization and content of the Egyptian Hieroglyph database.*

**Status**

*This is a **draft** document which may be updated, replaced, or superseded by other documents at any time. Publication does not imply endorsement by the Unicode Consortium. This is not a stable document; it is inappropriate to cite this document as other than a work in progress.*

**A Unicode Standard Annex (UAX)** forms an integral part of the Unicode Standard, but is published online as a separate document. The Unicode Standard may require conformance to normative content in a Unicode Standard Annex, if so specified in the Conformance chapter of that version of the Unicode Standard. The version number of a UAX document corresponds to the version of the Unicode Standard of which it forms a part.

Please submit corrigenda and other comments with the online reporting form [[Feedback](#)]. Related information that is useful in understanding this annex is found in Unicode Standard Annex #41, “[Common References for Unicode Standard Annexes](#).” For the latest version of the Unicode Standard, see [[Unicode](#)]. For a list of current Unicode Technical Reports, see [[Reports](#)]. For more information about versions of the Unicode Standard, see [[Versions](#)]. For any errata which may apply to this annex, see [[Errata](#)].

## Contents

- 1 [Introduction](#)
  - 2 [Mechanics](#)
    - 2.1 [Database Design](#)
    - 2.2 [Unikemet.txt](#)
  - 3 [Property Types](#)
    - 3.1 [Catalog Indexes](#)
    - 3.2 [Sources](#)
    - 3.3 [Description](#)
    - 3.4 [Function](#)
    - 3.5 [Core](#)
    - 3.6 [Mirroring and Rotation](#)
  - 4 [The Properties](#)
    - 4.1 [Alphabetical Listing](#)
    - 4.2 [Listing by Version of Addition to the Unicode Standard](#)
  - 5 [History](#)
  - [Acknowledgements](#)
  - [Modifications](#)
- 

## 1 Introduction

The Unikemet database is the repository for the Unicode Consortium’s collective knowledge regarding the Egyptian hieroglyphs contained in the Unicode Standard. It contains ancillary data to help implement support for the Egyptian hieroglyphs. (The term 'kemet' meant 'black land' in old Egyptian and was used as the official **name** of their country.)

Formally, Egyptian hieroglyphs are defined within the Unicode Standard via their names and assigned code points. However, while the first block: Egyptian Hieroglyphs (U+13000..U+1342F) has character names based on the Gardiner convention, the extended block: Egyptian Hieroglyphs Extended-A (U+13460..U+143FF) use algorithmic names of the type EGYPTIAN HIEROGLYPH-xxxxx where xxxxx is the **5-digit** hexadecimal value of the code point, therefore providing little information about the identity of the character. The ancillary data provided by the database define additional information such as a detailed description of the character, various sources, catalog entries, and function. It also defines properties related to these hieroglyphs, such as belonging to a Core set, whether they rotate or not, and whether **they** mirror or not.

This document is a guide to that data, describing the mechanics of the Unikemet database, the nature of its contents, and the status of the various properties.

## 2 Mechanics

### 2.1 Database Design

The database consists of a number of fields containing data for each Egyptian hieroglyph **s** in the Unicode Standard. The fields, all of which correspond to properties, have names that consist entirely of ASCII letters and digits with no spaces or other punctuation except for underscore. For historical reasons, they all start with a lowercase **k**.

All data in the Unikemet database is stored in UTF-8 using Normalization Form C (NFC). Note, however, that the “Syntax” descriptions below, used for validation of property values,

operate on Normalization Form D (NFD), primarily because that makes the regular expressions simpler.

## 2.2 Unikemet.txt

Included with the [UCD] is a file called `Unikemet.txt`. This is a snapshot of the public contents of the Unikemet database as of the release date for this version of the Unicode Standard.

The `zip` file is an archive of a single text file, in UTF-8, NFC, and using Unix line endings which contains the values for all properties in the Unikemet database. Properties are described by categories in this document but are nevertheless included in a single file (unlike, for example, the Unihan database).

In this file, blank lines may be ignored; lines beginning with `#` are comment lines used to provide the header and footer. Each of the remaining lines is one entry, with three, tab-separated fields: the Unicode Scalar Value, the property name, and the value for the property for the given Unicode Scalar Value. For most of the properties, if multiple values are possible, the values are separated by spaces. No hieroglyph may have more than one instance of a given property associated with it, and no empty properties are included in `Unikemet.zip`.

There is no formal limit on the lengths of any of the property values. Any Unicode characters may be used in the property values except for control characters (especially tab, newline, and carriage return). Note that unlike Unihan, double quotes are allowed but are discouraged and will likely be removed in a future version.

The data lines are sorted by Unicode Scalar Value and property type as primary and secondary keys, respectively.

The file's header includes a summary of the properties the file contains.

## 3 Property Types

The data in the Unikemet database serves a multitude of purposes, and the properties are most conveniently grouped into categories according to the purpose they fulfill. We provide here a general discussion of the various categories, followed by a detailed description of the individual properties, alphabetically arranged.

### 3.1 Catalog indexes

Two catalog indexes are defined: `KEH_Cat` and `KEH_UniK`. The catalog index `KEH_Cat` is defined using an IFAO-based sign taxonomy. It is written using a three-level classification: using a group index, a sub-group index, and an index within that sub-group. It uses a three-level classification. The higher level, the group index, is a combination of the Gardiner A-Z (and Aa) classification and the IFAO chapter classification (I to XXX in Roman notation). The second level uses the IFAO sub-chapter classification already present in the IFAO publication. The third level is a new index and just orders items sequentially within each sub-group. For example, the catalog index 'A-01-001' represents the first element, designated by 001, of the sub-group 'A-01'. The element 01 in 'A-01' represents the first sub-group of the group 'A'.

Within the group level, while IFAO may include a few more items, but these can be easily mapped into existing Gardiner groups. For example, the IFAO groupings (for example the Gods (Chapter III) and Goddesses (Chapter IV) can be

combined in the Gardiner group C (Anthropomorphic Deities). The following is the list of the first level groups and their relationship with the IFAO groups:

Gardiner groups	IFAO (translated from French)
A. Man and his occupations	I. Men and monarchs
B. Woman and her occupation	II. Women and monarchs
C. Anthropomorphic deities	III. Gods IV. Goddesses
D. Parts of the human body	V. Human body parts
E. Mammals	VI. Mammals
F. Parts of mammals	VII. Mammal body parts
G. Birds	VIII. Birds
H. Parts of birds	IX. Bird parts
I. Amphibious animals, reptiles, etc.	X. Reptiles, amphibians
K. Fishes and parts of fishes	XI. Fishes and parts of fishes
L. Invertebrate and lesser animals	XII. Insects and arachnids
M. Trees and plants	XIII. Plants
N. Sky, earth, water	XIV. Sky, earth, water
O. Buildings, parts of buildings, etc.	XV. Edifices and parts of edifices
P. Ships and part of ships	XVI. Boats and parts of boat
Q. Domestic and funerary furniture	XVII. <b>Everyday</b> and funeral furniture
R. Temple furniture and sacred emblems	XVIII. Temple furniture
S. Crowns, dresses, staves, etc.	XIX. Crowns XX. Jewels, clothes, staves
T. Warfare, hunting, butchery	XXII. Warfare, hunting, fishery, butchery
U. Agriculture, crafts, and professions	XXI. Agriculture and workshop tools
V. Rope, fiber, baskets, bags, etc.	XXIII. Rope, baskets, bags
W. Vessels of stone and earthenware	XXIV. Vases
X. Loaves and cakes	XXV. Bread loaves
Y. Writings, games, music	XXVI. Writings, games, music
Z. Strokes, signs derived from Hieratic, geometrical figures	XXVII. Geometric shapes
AA. Unclassified	XXVIII. Ill-defined signs

Notes:

- The order of the A-Z and XXVIII lists is identical, except for the two groups XXI and XXII (which correspond to the groups U and T, respectively).
- IFAO Chapter XXIX (Uncertain identity signs) and Chapter XXX (Conventional signs) are not used in the taxonomy because they are seldom used by other references.
- Some characters originally **in from** the group 'AA.XXVIII Unclassified Ill-defined signs' have been moved to other groups when their identity could be **confirmed/clarified**. Some members originally **in from** the IFAO group XXIX have also been reclassified.

Because this catalog number is still a work in progress, its status is provisional.

The `KEH_UniK` catalog index was originally defined exclusively for the original Unicode Egyptian Hieroglyph block and is part of the formal character name for these code points. This catalog index, it has been extended to cover all newly encoded signs. The code points which refer to the same. For example, the values common to Hieroglyphica and JSesh source value use the prefix are encoded in the form HJ followed by a space and the common value between Hieroglyphica and JSesh but zero padded to 3 digits and the values AA or Aa represented by AA. For example, the catalog index for U+1346C is HJ A072A indicating that the code point is associated with the same Hieroglyphica and JSesh source value: A72A. New entries not common to Hieroglyphica and JSesh were given new values without a prefix. The main rationale for the catalog index that source is to provide a Gardiner-like notation for all Egyptian hieroglyphs, which is a feature requested by desired feature for some Egyptologists. A significant issue is that the name space shared among the original Gardiner notation, the Unikemet original catalog index, Hieroglyphica and JSesh values has many collisions. For example, U+1304E has A71 as sources; the sources A71 for both Hieroglyphica and JSesh, but was assigned to A069 in the original block. In comparison, in counterpart, the U+1346A in the extended block has A69 as sources; the sources A69 for both Hieroglyphica and JSesh. To avoid an apparent name collision, the catalog index for this character it is not HJ A069, but A069A. Therefore, the notation 'HJ' is only used for new characters when the common Hieroglyphica and JSesh source values do not collide with `KEH_UniK` values used in the original block.

### 3.2 Sources

Sources are among the normative parts of the Unikemet database, and refer to some well-known Egyptian hieroglyphs collections. These sources are defined as `KEH_HG`, the Hieroglyphica classification, `KEH_JSesh`, the JSesh index, and `KEH_IFAO`, the IFAO entries. While these values are normative, they are not immutable. Some values may be a matter of interpretation or may contain as there are still some matter of interpretation and some errors. Many of these sources only use glyphic evidence, don't refer to the original paleographic attestations, and don't provide a formal description of the referred sign.

Detailed descriptions of the syntax used for these sources are to be found in [Section 4.1, Alphabetical Listing](#), below.

### 3.3 Description

While the description `KEH_Desc` is only informative, it is an essential part of the identity of an Egyptian hieroglyph. Because many attestations of these signs are imprecise, due to the imperfect preservation of the original evidence, Egyptologists had to come to a rough consensus on how to describe the abstract form of these signs as precisely as possible. While this description still allows variation in the font style used for their representation, it is expected that all these variants will adhere to the description as stated by this property. Due to the complexity of some of these signs, the description can be a rather long expression.

For example, the description for U+13A6E reads as follows following:

'A ram (*Ovis longipes palaeo-aegyptiacus*), standing, without a beard, with a cobra (*Naja haja*), standing up, with expanded hood (*Uraeus*)(I64) on its head, with the wings of a bird on its back, spread in a v-shape.'

Note that the description currently uses the Hieroglyphica/JSesh references in many of these descriptions to designate another sign included in the sign. The example above, 'I64' refer to U+13D79 which is itself described as 'A cobra (*Naja haja*), standing up, with

expanded hood (Uraeus)'. Because Hieroglyphica and JSesh do not always coincide, in case of differences, the JSesh reference prevails.

### 3.4 Function

The function type `KEH_Func` and its corresponding function value `KEH_FVa1` are **only provisional; provisional, only because** they are still a work in progress. All signs are expected to have a function type representing either a pictogram, a logogram, a phonogram (or "phonemogram" **the term phonemogram is also used**), a classifier (or **"determinative" also commonly called determinative**), phono-repeater, a radicogram, or an interpretant. **The function type also includes; and** a function value **with using** transliterated text.

The following text defines the function types:

- Pictogram – **p**ictorial symbol. It typically has no pronunciation.
- Logogram – sign that represents a word **in of** the Egyptian language. As such it has a pronunciation and a meaning.
- Phonogram or phonemogram – sign that represents a sound **in of** the writing system. It does not carry a semantic value. Strictly speaking, if we make a distinction between phonetics and phonology, the term phonemogram would be preferred to denote a **phonological phonology** concept, but the two terms tend to be used interchangeably.
- Classifier – sign written at the end of words that indicates the semantic category to which the respective word belongs. As such it is always mute. It is traditionally called **a** determinative.
- Phono-repeater – a sub **category** of classifier which has a phonetic meaning.
- Radicogram – graphemes that **both** point **at the same time** to some form and some content, but are not able **alone** to refer to an autonomous lexeme **alone**.
- Interpretant – non **autonomous** graphemes that interpret the phonemic values of other semograms or phonograms.

The function value uses the transliteration format convention **that is** commonly known as the Gardiner 1957 convention. **This convention** already **appears in the names list annotations; in use in the block annotations part** of the original Egyptian Hieroglyph block. **The transliteration format; it** uses the following letters: ʒ, □, y, ʿ, w, b, p, f, m, n, r h, h, h, h, s, š, k, k, g, t, t, d, d. It may **also** contain additional punctuation for optional part, alternative, semantic element, etc. **This will be developed in future versions of this document.**

### 3.5 Core

The normative property `KEH_Core` determines whether an Egyptian hieroglyph is part of a 'Core' set. The 'Core' set is a curated subset of characters from the full Egyptian hieroglyph encoded set. It is the recommended set for Egyptologists and should be implemented in widely used fonts. The Core set represents the opinion of experts **who reviewed the; based on presented** evidence **that was provided to them;** (The same group reviewed the full set. **)))**) This set is similar to UnihanCore2020 for CJK, which is the minimal set of required ideographs for East Asia. For a description of the selection process for the Core set by the Egyptologists involved, see the "Principles" Appendix. Characters in the Core set were verified by an image **in (photographs and trustworthy facsimiles);** Transcription (a hand-drawn sketch of a sign) alone was not normally considered to be verified evidence. Images from hieratic texts could be considered if the hieroglyphic nature of the sign could be easily reconstructed (cursive hieroglyphs).

The following are the exceptions to the requirement for verification:

- the sign appears in the Unicode 5.2 repertoire,
- the sign while it could not be verified and could, it could also not be constructed using an overlay or insertion mechanism.

While the property is normative, it is not immutable, that is, signs may move in or out of the Core set.

### 3.6 Mirroring and Rotation

The properties `KEH_NoMirror` and `KEH_NoRotate` indicate specific and rare behavior for some Egyptian hieroglyphs.

Most Egyptian hieroglyphs are expected to mirror relative to the reading direction. For example, for asymmetrical 'faces', the face is expected to face the start of the text, whether the line runs RTL or LTR. In very rare cases, the sign has a fixed orientation concerning mirroring. For example, U+130BB and U+130BD are an apparent set of mirrored walking legs. However, these two signs indicate opposite walking directions. In these rare cases, the property value `KEH_NoMirror` will be set to 'Y'.

Similarly, most Egyptian hieroglyphs can be rotated without changing their meaning. Because these rotations are a common occurrence, variation selectors `shouldcan` be used to represent these alternate representations. However, there are some signs where the rotation is significant and therefore they cannot be rotated. In these rare cases, the property value `KEH_NoRotate` will be set to 'Y'.

## 4 The Properties

We now give two listings of the properties in the Unikemet database. The first is an alphabetical listing, with information on the property contents and syntax. The second is a listing of the properties by the version of the Unicode Standard in which they were first introduced.

### 4.1 Alphabetical Listing

For each property we give the following information in the alphabetical listing: its *Property* tag, its Unicode *Status*, its *Category* as defined above, the Unicode version in which it was *Introduced*, its *Delimiter*, its *Syntax*, and its *Description*.

The *Property* name is the tag used in the Unikemet database to mark instances of this property.

The Unicode *Status* is either *Normative*, *Informative*, or *Provisional*, depending on whether it is a normative part of the standard, an informative part of the standard, or neither. We may also include *Deprecated* as a Unicode Status if the property is no longer to be used.

Most of the properties which allow multiple property values have a *Delimiter* defined as "space" (U+0020 SPACE). Properties which do not have multiple property values have this defined as "N/A." Some properties do not currently have multiple values in the data but may do so in the future.

For most properties with multiple values, the order of the values is arbitrary and has no particular significance. The most common order in such cases is alphabetical or numerical.

Because the property `KEH_Func` describing the function type may correspond to multiple types and may have also multiple values, the syntax is more complex. If there `are is`

multiple types, the types are separated by '/', but in most cases they share the same value. Multiple values are typically separated by either '/' or '|' (the "space" cannot be used because it may be part of a value field). Note that this is a work a progress, it denotes the current status among Egyptologist and may evolve over time. Note however that the vast majority of Egyptian hieroglyphs have a single function type and a single function value.

Validation is done as follows: The entry is split into subentries using the *Delimiter* (if defined), and each subentry converted to Normalization Form D (NFD). The value is valid if and only if each normalized subentry matches the property's *Syntax* regular expression. Note that any given property's *Syntax* is not guaranteed to be stable and may change in the future.

Finally, the *Description* contains not only a description of what the property contains, but also source information, known limitations, methodology used in deriving the data, and so on.

The properties covered in the table are: [kEH\\_Cat](#), [kEH\\_Core](#), [kEH\\_Desc](#), [kEH\\_Func](#), [kEH\\_FVal](#), [kEH\\_HG](#), [kEH\\_IFAO](#), [kEH\\_JSesh](#), [kEH\\_NoMirror](#), [kEH\\_NoRotate](#), and [kEH\\_UniK](#).

Property	<b>kEH_Cat</b>
Status	Informative
Category	Catalog Indexes
Introduced	16.0
Delimiter	N/A
Syntax	([A-IK-Z] AA)-[0-9]{2}-[0-9]{3}
Default	N/A
Description	Catalog entry corresponding to the IFAO-based taxonomy

Property	<b>kEH_Core</b>
Status	Normative
Category	Core
Introduced	16.0
Delimiter	N/A
Syntax	Y N
Default	N
Description	It determines whether an Egyptian hieroglyph is part of the 'Core' set.

Property	<b>kEH_Desc</b>
Status	Informative
Category	Description
Introduced	16.0
Delimiter	N/A
Syntax	[^\\t"]+
Default	N/A



Description	Detailed description of the appearance of the hieroglyph. It can be any Unicode character, except for control characters.
-------------	---

Property	<b>kEH_Func</b>
Status	Provisional
Category	Function
Introduced	16.0
Delimiter	/ (see description)
Syntax	[^\t"]+
Default	N/A
Description	All signs are expected to have a function type representing a pictogram, a logogram, a phonemogram (or "phonogram" the term phonogram is also used), a classifier (or "determinative" also commonly called determinative), a phone-repeater (sub-category of classifier), a radicogram or interpretant. It can be any Unicode character, except for control characters. Some types such as logogram have an English description, while others such as phonemogram typically do not. Most signs have a single type, but some have multiple types (separated by '/'). Sometimes additional context may be included in the type description, including transliterated text. This, this text can also use '/' to denote alternative description. Finally, while some signs are clearly attested, their type is uncertain, unknown, or undocumented as yet. That uncertainty is mentioned in the text itself.

Property	<b>kEH_FVal</b>
Status	Provisional
Category	Function
Introduced	16.0
Delimiter	/ or   (see description)
Syntax	[β□y'wbpfmnrhḥḥsškgttdd./-;()\s]+
Default	N/A
Description	All signs are expected to have a function value corresponding to their function type. The value is expressed using the Gardiner 1957 convention for Egyptian hieroglyph transliteration. The delimiters '/' or ' ' are used to separate alternative values, while other punctuations may represent syntax elements, optional values, etc. The current value field represents a draft version, as work is still in progress and will be refined, based on feedback. Some signs still do not have a function value but are expected to be documented in the future.

Property	<b>kEH_HG</b>
Status	Normative
Category	Sources
Introduced	16.0
Delimiter	space
Syntax	([A-IK-Z] AA)[0-9]{1,3}[A-Z]{1,2}  US
Default	N/A

Description	Hieroglyphica source as specified in <i>Hieroglyphica – Sign List</i> , Nicholas Grimal, Jochen Hallof, Dirk van der Plas, 2nd edition, 2000. Multiple Hieroglyphica entries could be assigned to the same code point.
-------------	--

Property	<b>kEH_IFAO</b>
Status	Normative
Category	Sources
Introduced	16.0
Delimiter	space
Syntax	[0-9]{1,3},[0-9]{1,2}
Default	N/A
Description	IFAO source value defined as page number and order in that page, separated by a comma. IFAO is defined as <i>Catalogue de la fonte hiéroglyphique de l'imprimerie de l'I.F.A.O.</i> , Institut Français d'Archéologie Orientale du Caire, 1983, IF607, SEVPO, Paris, France. Multiple IFAO entries could be assigned to the same code point.

Property	<b>kEH_JSesh</b>
Status	Normative
Category	Sources
Introduced	16.0
Delimiter	space
Syntax	([A-IK-Z] Aa NL NU)[0-9]{1,3}[A-Za-z]{1,4}  (US1 US22 US248 US685)([A-IK-Z] Aa NL NU)[0-9]{1,3}[A-Za-z]{1,4}
Default	N/A
Description	JSesh source as specified in Rosmorduc, Serge. (2014). JSesh Documentation. [Online, version 7.5.5] Available at: <a href="http://jseshdoc.qenherkhopeshef.org">http://jseshdoc.qenherkhopeshef.org</a> [Accessed Feb 23rd 2021]. Current version is 7.6.1 as of October 4th 2023, and sources values may have to be updated accordingly. Multiple JSesh entries could be assigned to the same code point.

Property	<b>kEH_NoMirror</b>
Status	Normative
Category	Mirroring and Rotation
Introduced	16.0
Delimiter	N/A
Syntax	Y N
Default	N
Description	It determines whether an Egyptian hieroglyph does not mirror. Note the reverse property because by default, most hieroglyphs can be mirrored depending on the reading direction.

Property	<b>kEH_NoRotate</b>
Status	Normative
Category	Mirroring and Rotation

Introduced	16.0
Delimiter	N/A
Syntax	Y N
Default	N
Description	It determines whether an Egyptian hieroglyph does not rotate. Note the reverse property because by default, most hieroglyphs can be rotated without affecting their meaning.

Property	<b>kEH_UniK</b>
Status	Provisional
Category	Catalog Indexes
Introduced	16.0
Delimiter	N/A
Syntax	([A-IK-Z] AA NL NU)[0-9]{3}[A-Z]{1,2}   HJ ([A-IK-Z] AA)[0-9]{3}[A-Z]{1,2}
Default	N/A
Description	Original Unikemet catalog index used by the Egyptian Hieroglyph block, augmented for the extended blocks. Note that this is a work in progress with some issues.

## 4.2 Listing by Version of Addition to the Unicode Standard

The table below lists the properties of the Unihan database by the version of the Unicode Standard in which they were first added.

Version	Properties Added	Properties Removed
16.0	<a href="#">kEH_Cat</a> , <a href="#">kEH_Core</a> , <a href="#">kEH_Desc</a> , <a href="#">kEH_Func</a> , <a href="#">kEH_FVal</a> , <a href="#">kEH_HG</a> , <a href="#">kEH_IFAO</a> , <a href="#">kEH_JSesh</a> , <a href="#">kEH_NoMirror</a> , <a href="#">kEH_NoRotate</a> , <a href="#">kEH_UniK</a>	

## 5 History

The Unikemet database originated as a concept proposed by the original Egyptian Hieroglyph proposal (ISO/IEC JTC1/SC2/WG2 N3237 ~~aka~~ L2/07-097) as an appendix to that document but never materialized as a true dataset. It contained original source references which have been partly superseded by this version. It should also be noted that N3237 is not 100% identical to what was eventually adopted by ISO and Unicode and was not updated to reflect the final code point values.

## References

For references for this annex, see Unicode Standard Annex #41, “[Common References for Unicode Standard Annexes](#).”

## Acknowledgements

This new database is the result of a collective work by many Egyptologists and is still a work in progress.

## Modifications

### Revision 2

- **Draft** of the first version of UAX #57 for Unicode 16.0.0.
- Fixed various typos in the document and made other small edits.

### Revision 1

- **Proposed Draft** of the first version of UAX #57 for Unicode 16.0.0.
- Split original kEH\_Func into a description/type still called kEH\_Func and a new property: kEH\_FVal expressing the function transliterated value.
- Fixed various typos in the kEH\_Func description.

Previous revisions will be accessed with the “Previous Version” link in the header when appropriate.

---

© 2024 Unicode, Inc. All Rights Reserved. The Unicode Consortium makes no expressed or implied warranty of any kind, and assumes no liability for errors or omissions. No liability is assumed for incidental and consequential damages in connection with or arising out of the use of the information or programs contained or accompanying this technical report. The Unicode [Terms of Use](#) apply.

Unicode and the Unicode logo are trademarks of Unicode, Inc., and are registered in some jurisdictions.