Traditional Mongolian Unicode Proposal (z52)

Chapter 1, Historical Background

1. Historical Origins of the Traditional Mongolian Script

The Traditional Mongolian script has a long and complex history, shaped by multiple stages of development, revision, divergence, and cultural interaction. Its evolution reflects sustained contact among Central Asia, East Asia, the Silk Road civilizations, and later imperial administrations.

The earliest form of the Traditional Mongolian script was derived from the Sogdian script, dating to approximately the 4~7th century CE. Sogdiana, located at the heart of the Silk Road connecting East Asia, Central Asia, West Asia, and parts of Europe and India, played a critical role in cultural and commercial exchange. Sogdian merchants, influenced heavily by Persian writing traditions, disseminated early script prototypes across these regions.

With the rise of Islamic power in Central Asia during the 7th century, many Sogdian merchants migrated eastward. A significant number entered Tang-dynasty China, particularly around the Dunhuang and Chang'an regions. The Tang dynasty, founded by descendants of the Xianbei tribes of northern origin, maintained strong openness toward Central Asian cultural influences. These migrations also coincided with the transmission of Buddhism into East Asia, evidenced by monumental Buddhist sites such as Dunhuang Caves (Mogao) and Xiuding Si (Henan).

During the late Tang period, portions of these populations moved northward through established trade networks with nomadic groups. Through these interactions, the Sogdian-derived script reached the Mongolian plateau.

---

2. Adoption by the Mongol Empire

Prior to the 13th century, multiple writing systems existed among the peoples of the Mongolian region. Several tribes adopted variations of the Sogdian-derived script. One such group was the Naiman tribe, which was conquered by Genghis Khan in 1204 after Battle of Chakirmaut.

Following this conquest, Genghis Khan ordered the Naiman scribe Tatatonga to formalize and teach the script at the imperial court. The script was adopted as the official writing system of the Mongol Empire and was taught to Genghis Khan's sons and grandsons. At this stage, the script consisted of approximately 18 basic letters and followed a relatively simple vowel–consonant grammatical structure.

---

3. Tibetan Influence and the Development of Vowel Harmony

In the 1240s, following the submission of Tibet to the Mongol Empire, the grand master Sakya Pandita Kunga Gyeltsen advised Prince Godan, a grandson of Genghis Khan, on linguistic and administrative matters. Under Tibetan scholarly influence, additional grammatical structure was introduced into the Mongolian script.

At this stage, the concept of vowel harmony was systematized, distinguishing:

- back vowels (yang vowels),

- front vowels (yin vowels),
- neutral vowel.

Despite this grammatical expansion, the vowel inventory remained limited to five vowels: *a, e, i, o, ü*. Traditional accounts attribute the conceptual distinction of vowel forms to Tibetan aesthetic and symbolic interpretations; however, in modern linguistic terms, this development represents a formalization of front–back vowel contrast rather than a gender-based classification.

---

4. Expansion During the Yuan Dynasty

In 1311, Choijin_Osor, a descendant of Kunga Gyeltsen and tutor to Khayishan Külüg Khan of the Yuan dynasty, further reformed the script. These reforms expanded the vowel system to seven vowels, structured as:

- front vowels: *e, ö, ü*
- back vowels: *a, o, u*
- neutral vowel: *i*

This structure closely resembles the vowel harmony system recognized in the modern Traditional Mongolian script.

---

5. Post-Yuan Fragmentation and Regional Divergence

Following the collapse of the Yuan dynasty, Mongols retreated from China back to the Mongolian plateau. By this time, other successor states of the Mongol Empire, such as the Golden Horde and the Ilkhanate, had adopted Islam and increasingly used Arabic-based scripts (including Chagatai), further diversifying Mongolic writing traditions.

Within Mongolia, political fragmentation led to multiple regional education systems and script traditions. Due to the absence of centralized linguistic authority, the Traditional Mongolian script diverged into numerous regional variants.

Among these, the Western Mongols (Oirat), under the leadership of Zaya Pandita, developed the Todo script, while Eastern Mongols continued using older forms with localized variations. This divergence resembles multiple parallel branches evolving independently, resulting in substantial complexity for later standardization efforts.

In the northeastern regions, Avandari developed a specialized Mongolian script for the Buryat population, which later became part of the Russian sphere. Each regional group introduced new letters, orthographic conventions, and grammatical rules.

---

6. Qing Dynasty Influence and Script Interference

The Eastern and Central Mongols were later incorporated into the Qing dynasty. The Manchu script, itself derived from Mongolian script but with structural modifications, became the official administrative language of the Qing government.

As official documents were written primarily in Manchu, Manchu grammatical and orthographic features influenced Mongolian writing. This led to increased ambiguity in letter usage (e.g., interchangeability of *T/D*, *H/G* in certain contexts) and further divergence among regional Mongolian scripts.

By this period, multiple branches of the Mongolian script coexisted, each reflecting distinct pronunciation systems and orthographic logic.

At this time, with massive amount translations of tibeten books into Mongolian, the Isolate set of approximately 18 letters gradually expanded to roughly 32 distinct letters.

---

7. Increasing Structural Complexity

Historically, Mongolian education emphasized a phonetic teaching system with initial, medial, and final letter forms. However, the coexistence of hard vowels, soft vowels, neutral vowels, consonant-only contexts, and vowel-alternation rules meant that even initial forms could have multiple variants. Medial and final forms exhibited even greater variation.

---

8. Twentieth-Century Reforms and Cyrillic Influence

In the 20th century, following Mongolia's independence, the government—under strong Soviet influence—mandated the adoption of the Cyrillic script. As a consequence, large segments of the Mongolian population lost active literacy in the Traditional Mongolian script.

When efforts were later made to revive Traditional Mongolian writing, Cyrillic-based grammatical concepts were often projected back onto the traditional script, introducing additional inconsistencies.

---

9. Contemporary Usage and Regional Variation and Chinese influence

In contrast, Mongolians in Inner Mongolia (under Chinese administration) continued to use the Traditional Mongolian script. Nevertheless, regional pronunciation differences led to variations in phonetic education methods, even though printed written forms remained relatively unified.

For inner Mongolians, with the influence from Chinese cultures, 3 more letters, zhi, chi, shi, was added from Chinese pronunciation, thus make it full 35 pronunciation letters.

---

10. Computerization and Unicode Standardization

Beginning in the 1980s, scholars from China, Mongolia, Germany, Japan, and other regions began developing computer-based encoding systems for Mongolian script. Due to technological limitations and region-specific linguistic assumptions, none of these early systems achieved global applicability.

In 2000, the Mongolian script was incorporated into the Unicode Standard. However, due to unresolved disagreements stemming from historical divergence, cultural influences (Sogdian, Tibetan, Russian, Chinese), regional pronunciation differences, and political considerations, only 35 basic letters were formally encoded.

Contextual letter forms (initial, medial, final variants) were not explicitly encoded and were instead delegated to font-level implementation under the Unicode model.

As a result, many historically and linguistically rooted disagreements were transferred from the Unicode standard layer to font-level, user-defined behavior.

---

11. Structural Consequence for Unicode

In practical terms, the current Mongolian Unicode ecosystem consists of two interdependent layers:

1. **Unicode-level encoding**, defining a limited set of abstract characters
2. **Font-level user agreement**, governing glyph selection, contextual shaping, and regional variation, with grammar elements imbedded.

This dual-layer structure reflects the historical complexity of the Traditional Mongolian script and remains a central challenge for comprehensive standardization.

Chapter 2: Problem Statement

2.1 Fragmentation of Font-Level Agreements

Due to historical, regional, and scholarly divergence, consensus among Mongolian user communities and academic groups has been difficult to achieve. As a result, multiple **font-level user agreements** have emerged, often operating in parallel and in conflict with one another.

Within Inner Mongolia, several competing font-level standards have been proposed. The most prominent include the **Choijinjab national standard (2010)** and the **GongXiang national standar (2023)** led by Professor Liang Jinbao. For an extended period, these implementations overlapped and overwrote each other in practical usage. More recently, Professor Liang's GongXiang standard has gained increasing institutional and financial support and is gradually becoming the dominant implementation.

In Mongolia proper, two principal groups have influenced font-level practice: the government-supported **Narangerel group**, and the technologically advanced **Bolor-Toli group** led by Badral. The Bolor-Toli group has proposed introducing an additional consonant letter to distinguish male and female variants of the consonant *G*, further diverging from other regional practices.

Because software applications developed by these groups are not internally synchronized, users frequently encounter incompatible font formats, inconsistent glyph selection, and divergent rendering behavior across operating systems and devices. These inconsistencies are a direct consequence of unresolved disagreement at the font-level agreement layer.

2.2 Unicode Dual-Layer Technical Limitations

A second major challenge arises from the **dual-layer structure** established when the Mongolian script was incorporated into the Unicode Standard in 2000. Under this model, Unicode defines a limited set of abstract characters, while contextual shaping and positional behavior are delegated to font-level technologies.

Modern computing environments vary significantly in platform architecture, rendering engines, and application frameworks. Many systems do not fully support advanced OpenType features. As a result, users and developers often encounter only Unicode-level base characters, displayed without the required initial, medial, and final forms that are essential to correct Mongolian orthography.

Although some operating systems—most notably Microsoft Windows—provide relatively comprehensive support for advanced OpenType shaping, many other platforms offer partial or inconsistent implementation. Correct rendering of the Traditional Mongolian script frequently depends on font-level mechanisms such as:

- Contextual alternates (`calt`)
- Chained contextual substitutions (`calt`, `ccmp`, `rlig`)
- Kerning and contextual positioning (`kern`, `GPOS`)
- Single glyph substitutions (GSUB Single Substitution)
- Multiple glyph substitutions (GSUB Multiple Substitution)
- Ligature substitutions (`liga`, `rlig`)

In environments lacking consistent support for these features, reliable display and processing of Mongolian text cannot be guaranteed.

2.3 Linguistic Ambiguity and Data Integrity

A third critical issue concerns **linguistic ambiguity and data integrity** in digital Mongolian text. In the current Unicode model, identical base letters are used to represent distinct vowel pairs such as *ö/ü* and *o/u*. This leads to frequent misencoding, incorrect character selection, and widespread ambiguity in digital text.

Similar ambiguity arises from mixed usage of consonant pairs such as *T/D* and *H/G*. Also, letter K was encoded twice, 1 for Mongolia users, and 1 for china users. These inconsistencies negatively affect search accuracy, text normalization, and corpus reliability. More importantly, they significantly degrade the quality of datasets used in artificial intelligence and natural language processing, where precise and stable character distinctions are essential.

As a consequence, organizations and developers are often required to employ specialized teams to manually identify and correct misencoded characters in large datasets. This process is labor-intensive, costly, and inherently unsustainable.

2.4 Sustainability, Cost, and Ecosystem Stability

The cumulative effect of fragmented font-level agreements, platform-dependent rendering behavior, and persistent linguistic ambiguity has resulted in long-term sustainability challenges for the Mongolian digital ecosystem.

Maintaining compatibility across fonts, applications, and platforms requires continuous manual intervention, repeated data correction, and customized software adaptation. These efforts impose significant financial and operational costs on institutions, developers, and end users.

Furthermore, the reliance on advanced font-level technologies—combined with inconsistent platform support—creates structural instability. Applications must either restrict functionality to the lowest common denominator or accept unpredictable rendering behavior across environments.

While the Unicode dual-layer model has enabled the basic use of the Traditional Mongolian script in digital systems, its current form places a disproportionate burden on font developers, application vendors, and data maintainers. Without improved standard-level clarity or structural refinement, achieving long-term interoperability, data reliability, and ecosystem stability will remain difficult.

Chapter 3: Proposed Approach

This proposal recommends the establishment of a **secondary, modernized Unicode encoding model for the Traditional Mongolian script**, designed to coexist with the current Unicode encoding. This parallel model would allow users, developers, and scholars to adopt the encoding most suitable to their needs, while enabling gradual convergence toward a unified, modern Unicode standard that more accurately reflects real-world Mongolian usage.

Consolidation of the Dual-Layer Model

The current dual-layer architecture—Unicode-level character encoding combined with extensive font-level shaping—is proposed to be consolidated into a **single-layer Unicode encoding model**. In practice, Mongolian users operate directly with initial, medial, and final letter forms. Therefore, this proposal recommends encoding these positional forms explicitly at the Unicode level, rather than deriving them indirectly through font-level contextual substitution.

By encoding the positional forms directly, the system can bypass divergent font-level user agreements and instead encode the intended orthographic outcome directly, thereby "starting with the end in mind."

Reduction of Redundant and Ambiguous Letter Usage

This proposal further recommends reducing ambiguity by unifying duplicated or conflated letter representations. Specifically, historically distinct vowel pairs such as *ö/ü* and *o/u*, as well as consonant distinctions involving *k*, *t/d*, and similar glyph substitutions, should be normalized into clearly defined Unicode characters. This approach would eliminate ambiguity arising from glyph swapping and inconsistent character usage.

Reduced Dependence on Font-Level Technologies

By encoding positional forms directly, the proposed model significantly reduces reliance on advanced OpenType font technologies. Complex dependencies on contextual alternates, chained substitutions, and glyph-

based disambiguation would be minimized, allowing Unicode characters themselves to represent the written forms actually used by readers and writers.

Reconstruction Based on Historical Structural Elements

The proposal supports reconstructing Mongolian letters by analyzing their historical structural components and decomposing them into elemental forms. These elements can then be recombined to form standardized initial, medial, and final representations within the Unicode encoding framework.

Delegation of Variation Management to Input Method Editors (IMEs)

Rather than embedding grammatical, regional, phonetic, or scholarly variation into the Unicode encoding itself, this proposal recommends managing such variation at the **input method editor (IME) layer**. Different IMEs may implement distinct grammatical rules, regional pronunciations, or academic conventions according to user preference.

Crucially, regardless of the IME used, the resulting encoded text would converge on a **unified Unicode representation**, ensuring that stored data remains consistent, searchable, interoperable, and maintainable. This approach enables diversity at the input stage while preserving uniformity at the encoding and database levels, significantly reducing long-term maintenance costs and improving sustainability.

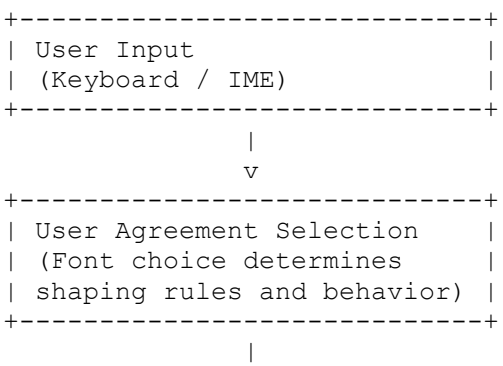
This proposal advocates a parallel, modernized Unicode encoding model for the Traditional Mongolian script that consolidates positional forms at the Unicode level, reduces ambiguity, minimizes font-level dependency, and delegates variation handling to IMEs, thereby improving interoperability, data integrity, and long-term sustainability.

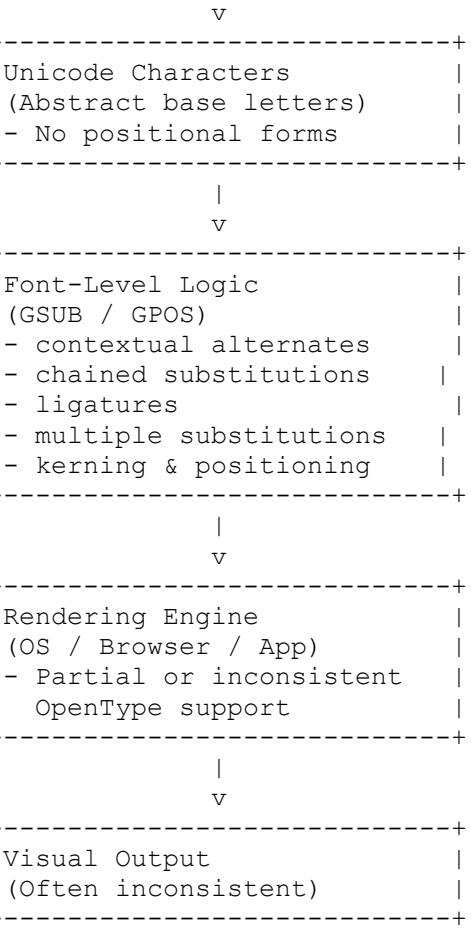
Chapter 4 Architecture Overview: AS-IS vs TO-BE

---

1. AS-IS Architecture (Current Unicode Mongolian Model)

Conceptual Structure





AS-IS Characteristics

- Unicode encodes **only abstract base characters**
- Initial, medial, and final forms are **not encoded**
- Actual written forms are resolved **indirectly**
- Font selection implicitly determines:
  - grammatical interpretation
  - vowel harmony behavior
  - glyph disambiguation
- The effective standard is therefore:

Unicode + Font + User Agreement

Key Observations

1. User Agreement Is an Undocumented Control Layer

The choice of font implicitly selects a **user agreement**, defining how abstract Unicode characters are transformed into visible text. This agreement is:

- not standardized
- not machine-readable
- not portable across platforms
- not visible at the Unicode level

As a result, the same Unicode sequence may represent different linguistic meanings depending solely on font choice.

---

2. Unicode Characters Do Not Represent Written Reality

Although Mongolian users read and write using **positional forms**, these forms are not represented in the Unicode character stream. Instead, they are reconstructed later through font logic, making the encoded data incomplete and ambiguous when separated from its rendering context.

---

3. Font-Level Logic Becomes a De Facto Standard

Because font-level GSUB/GPOS rules determine actual behavior, font implementations effectively function as **unofficial standards**, leading to:

- competing implementations
- incompatibility between applications
- inconsistent rendering across platforms

---

4. Rendering Is Platform-Dependent

Correct output depends on the rendering engine's support for advanced OpenType features. Where such support is incomplete or inconsistent, the script fails to render correctly, despite the Unicode text itself being valid.

AS-IS Failure Points (Explicit)

1. Missing OpenType support → broken text
2. Conflicting font standards → incompatible output
3. Same Unicode sequence → different visual result
4. Databases store ambiguous text
5. AI / search systems cannot reliably disambiguate

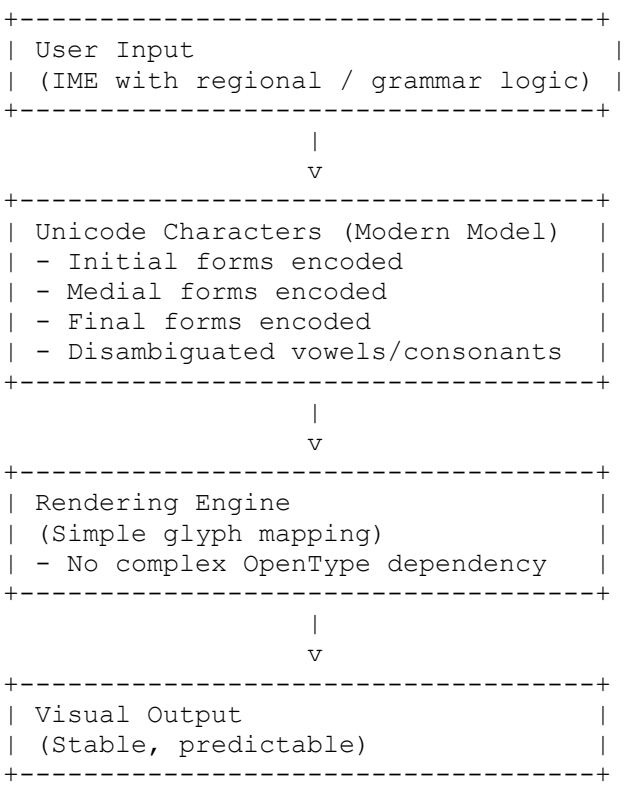
---

3. TO-BE Architecture (Proposed Unified Unicode Model)

The TO-BE architecture is derived by removing implicit behavior from the font layer**,** eliminating user agreement selection via fonts**,**

and restoring Unicode as the authoritative representation of written forms**, while preserving IME flexibility.**

Conceptual Structure



TO-BE Characteristics

- Unicode encodes **actual written forms**
- Initial / medial / final forms are **explicit**
- Font logic is simplified
- Grammar, regional, scholarly variation handled by **IME**
- Stored text is:
    - unambiguous
    - searchable
    - AI-ready
    - platform-independent

---

3. AS-IS vs TO-BE (Side-by-Side Comparison)

Aspect	AS-IS (Current)	TO-BE (Proposed)
Unicode role	Abstract base letters	Concrete positional forms
Initial/Medial/Final	Font-derived	Unicode-encoded
Font complexity	Very high	Minimal
OpenType dependency	Critical	Optional
Platform consistency	Low	High
User agreement	Fragmented	Centralized
Data integrity	Weak	Strong
Search / AI accuracy	Degraded	Reliable
Maintenance cost	High	Low
Long-term sustainability	Poor	Strong

Architectural Changes (One-to-One with AS-IS)

AS-IS Layer	TO-BE Change
Font choice selects user agreement	User agreement moved to IME
Unicode encodes abstract letters	Unicode encodes written forms
Font performs grammatical logic	Font performs rendering only
Rendering depends on OpenType	Rendering uses simple glyph lookup
Same text renders differently	Same text renders identically

This architecture does **not** remove IME freedom or historical variation; it **relocates responsibility to the correct layer**.

Layer-Violation Analysis

Current Violations in AS-IS Architecture

1. Unicode Layer Violation

Unicode's role is to encode characters representing written language.

Current violation:

- Unicode encodes abstract base letters
- But users write and read positional forms
- Unicode therefore does *not* encode the written reality

Result:
Unicode data is incomplete without font context.

---

2. Font Layer Violation

Font's role is glyph selection and visual rendering.

Current violation:

- Fonts implement grammatical rules
- Fonts encode vowel harmony logic
- Fonts resolve phonological ambiguity
- Fonts act as de facto standards

Result:
Fonts become **linguistic authorities**, which violates the Unicode architecture model.

---

3. IME Layer Violation

IME's role is input interpretation and user intent.

Current violation:

- IMEs emit ambiguous Unicode sequences
- IMEs defer interpretation to fonts
- IMEs cannot guarantee semantic correctness

Result:
IME logic is weakened, while fonts assume responsibilities they cannot reliably fulfill.

Summary of Violations

Layer	Intended Responsibility	Actual Responsibility
IME	Grammar & variation	Reduced / bypassed
Unicode	Written representation	Abstract placeholders
Font	Visual rendering	Grammar & semantics

This inversion of responsibility is the **core structural failure**.

Reviewer-Facing Justification: Why the AS-IS Architecture Fails at Scale

This section is written **explicitly for Unicode reviewers**.

---

1. Non-Deterministic Text Representation

Under the current architecture, the same Unicode sequence may produce different visual and linguistic results depending on:

- font selection
- platform rendering engine
- OpenType feature support

This violates the expectation that Unicode text should have **stable interpretation independent of presentation**.

---

2. Implicit Standards Cannot Scale

Font-based user agreements function as **implicit, undocumented standards**. Such agreements:

- are not discoverable
- are not enforceable
- are not interoperable
- cannot be validated by software

This model may function in limited environments but **cannot scale globally**.

---

3. Platform Fragmentation Is Inevitable

Advanced OpenType shaping is unevenly supported across:

- operating systems
- browsers
- mobile platforms
- embedded systems

A Unicode model that requires full OpenType support to function correctly **will always fail on some platforms**.

---

4. Data Integrity and AI Readiness

Modern computing increasingly relies on:

- search indexing
- corpus analysis
- machine learning
- AI training pipelines

Font-dependent semantics are invisible to these systems. As a result:

- text becomes ambiguous in databases
- normalization is unreliable
- AI models learn corrupted data

This is not a theoretical concern; it is already observable in production systems.

---

5. Maintenance Cost Grows Non-Linearly

As the ecosystem grows:

- each new font multiplies incompatibility
- each new platform multiplies failure points
- each dataset requires manual correction

This creates a **non-linear cost curve**, making long-term sustainability impractical.

Final Reviewer-Facing Statement

The current Unicode Mongolian architecture embeds linguistic interpretation within font behavior, relies on implicit user agreements, and assumes uniform platform support for advanced shaping technologies. While functional in constrained environments, this model does not scale across platforms, datasets, or modern computational use cases. A Unicode-centric representation of written forms, with variation handled at the input layer, restores architectural correctness and enables long-term interoperability.

---

4. Key Architectural Benefits (Unicode-Relevant)

4.1 Separation of Concerns (Correct Layering)

- **Unicode**: stable, unambiguous representation
- **IME**: grammar, pronunciation, academic variation
- **Fonts**: simple glyph rendering

This restores **architectural correctness**.

4.2 Elimination of Font-Level Standard Wars

- No need for competing font agreements
- No glyph swapping logic
- No platform-specific shaping hacks

4.3 Data-First Design (Search & AI Safe)

- One encoded form = one meaning
- Corpora become reliable
- Machine learning pipelines improve immediately

4.4 Backward Compatibility Strategy (Implicit)

- AS-IS model continues unchanged
- TO-BE model coexists as **secondary standard**
- Gradual adoption via IMEs and applications

In the current architecture, Traditional Mongolian text rendering depends on a combination of abstract Unicode characters, implicit user agreements expressed through font selection, and platform-specific font shaping support, resulting in inconsistent visual output and ambiguous encoded data.

Chapter 7: Proposed Code Points and Encoding Model

7.1 Rationale for Additional Code Points

As this proposal requests the allocation of additional Unicode code points, it is necessary to define a clear and systematic encoding model that reflects the structural reality of the Traditional Mongolian script.

Rather than encoding only abstract base letters and delegating written forms to font-level behavior, this proposal introduces a set of **elemental Unicode characters** that can be combined to reconstruct the full range of historically attested Mongolian letter forms. These elements correspond to recurring structural components observed across initial, medial, and final positions.

Under this model, Mongolian letters are not treated as indivisible units, but as **compositions of standardized elements**, allowing precise and unambiguous representation of written forms.

Architectural Model of the Zcode for Mongolian Script

The Traditional Mongolian script can be understood as a **layered and compositional system**, in which a small set of phonetic letters expands into a rich set of visual forms, all of which can be represented in a stable and deterministic way.

```
Mongolian Script
 └── Letters (35 core letters, as defined by phonetic education theory)
     └── Positional Forms (4 per letter)
         ├── Isolate (reference form)
         ├── Initial (word-initial form)
         ├── Medial (word-medial form)
         └── Final (word-final form)
             └── Variants (one or more per position)
                 └── ZCode Components
                     (one or more ZCode characters per variant)
```

Quantitative Overview

- **35 core letters**
  Defined by traditional Mongolian phonetic education theory.
- **4 positional forms per letter**
  Isolate, Initial, Medial, and Final.
- **176 total visual variants**
  When all positional forms and their variants are taken together, the entire Mongolian script consists of **176 distinct letter variants**.
- **52 ZCode components**
  All 236 variants can be constructed using **only 52 ZCode code points**, which serve as the smallest reusable structural elements.

Conceptual Flow

Using this architecture, the representation of Mongolian letters proceeds in three clear stages:

1. **Phonetic abstraction (35 letters)**
   The script begins with a compact, phonetic model suitable for education and linguistic analysis.

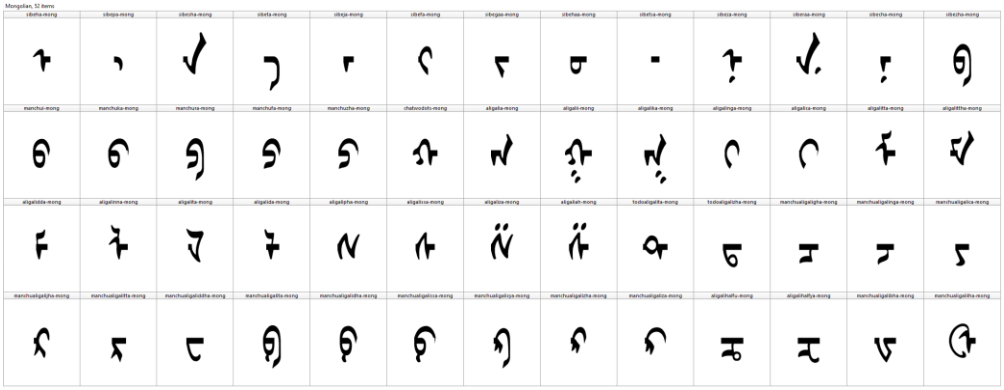
2. **Positional and variant expansion (176 variants)**
   Each letter expands into multiple positional forms, and each positional form may have several valid variants, reflecting traditional writing practice.
3. **Component-based realization (52 ZCode components)**
   Every variant is constructed from one or more ZCode components, allowing all Mongolian letter shapes to be represented **accurately, consistently, and deterministically**.

Key Insight

A complex writing system of 176 distinct Mongolian letter variants can be fully and reliably represented using just 52 reusable ZCode components, while remaining faithful to the traditional 35-letter phonetic framework.

7.2 Proposed Unicode Element Set

This proposal introduces a total of **52 new Unicode code points**, each representing a distinct structural element used in the construction of Traditional Mongolian letters.



No.	Section	ZCode Name	ZCode Character	Notes	Temp Unicode in Font
Core Stroke and Structural Components					
1	Core Stroke	ti t em		stroke head	U+1865
2	Core Stroke	odoi _s uul		tail mark	U+1866
3	Core Stroke	or hi c h		hook / ending	U+1867
4	Core Stroke	vr vgs i l ga_or_chachi l ga /		terminal connector	U+1868
5	Core Stroke	achvg		vertical stroke	U+186A
6	Core Stroke	hvms		dot / i-vowel	U+186B
7	Core Stroke	si l bi		medial connector	U+186C
8	Core Stroke	gedes		rounded stroke	U+186D
9	Core Stroke	nur uu		backbone / extended tail	U+186E
Nasal Components					
10	N	n_titem		nasal head	U+186F
11	N	n_orhich		nasal ending	U+1870
12	N	n_achvg		nasal medial	U+1871
B / P Components					
13	B	b_banbvr_egsilge		b final form	U+1872
14	B	b_banbvr_baga		b base / small	U+1873
15	B	b_banbvr_ih		b extended	U+1874
16	P	p_chihi_egsilge		p final form	U+1875
17	P	p_chihi_baga		p base	U+1876
18	P	p_chihi_ih		p extended	U+1877
H / G Components					
19	H / G	h_hanhar		h base	U+1878
20	H / G	agsilga		h/g final	U+1887
21	H / G	g_hanhar		g base	U+1888

22	H / G	dvsaltai_agsilga		g final alternate	U+1889
23	H / G	hinhir		h/g initial variant	U+188A
24	H / G	hunhur		h/g medial variant	U+188B
M Components					
25	M	m_titem		m base	U+188C
26	M	m_hurbelge		m final	U+188D
27	M	m_achvg		m medial	U+188E
L Components					
28	L	l_titem		l base	U+188F
29	L	l_hurbelge		l final	U+1890
30	L	l_achvg		l medial	U+1891
S / X Components					
31	S / X	s_agshilga		s final	U+1893
32	S / X	s_sechig		s base	U+1894
33	S / X	x_agshilga		x final	U+1896
34	S / X	x_sechig		x base	U+1897
T / D Components					
35	T / D	t_senj		t base	U+1898
36	T / D	d_senj		d base	U+1899
Affricates / Approximants（塞擦音 / 近音）					
37	Affricates / Approximants	q_acha		q / ch base	U+189A
38	Affricates / Approximants	j_bosoo_silbi		j base	U+189B
39	Affricates / Approximants	y_erteger_silbi		y base	U+189C
40	Affricates / Approximants	r_Odoi_suul		r final	U+189D
41	Affricates / Approximants	r_erbgeljin_silbi		r base	U+189E
42	Affricates / Approximants	w_gohtai_silbi		w base	U+189F
Loan / Extended Letters					
43	Loan / Extended	f_chihi_egsilge		f final	U+18A0
44	Loan / Extended	f_chihi_baga		f base	U+18A1
45	Loan / Extended	f_chihi_ih		f extended	U+18A2
46	Loan / Extended	k_egsilge		k final	U+18A3
47	Loan / Extended	k_baga		k base	U+18A4
48	Loan / Extended	k_ih		k extended	U+18A5
49	Loan / Extended	c_acha		ts / c base	U+18A6
50	Loan / Extended	z_acha		z base	U+18A7
Galig Extensions					
51	Galig	galig_tvvrai		Galig base	U+18A8
52	Galig	galig_r_malgai		Galig modifier	U+18AA

7.3 Mapping table from Existing Unicode to Zcode Proposal

Mapping tables from legacy Mongolian Unicode Standard to Zcode Composition

No	Unicode Standard (As-Is)	Letter Group	Variant	Position	Zcode Composition (To-Be)	Zcode Component Names	Zcode Unicode Temp
— Vowels（元音）- Letter A							
1	U+1820 + FVS3	A	A-1	Isolate_0	ᠵ+᠋	titem + orhich	U+1865 + U+1867
2	U+1820 + FVS1	A	A-2	Isolate_1	ᠵ+ᡂ	titem + vrvgsilga_or_chachilga	U+1865 + U+1868
3	U+1820 + FVS2	A	A-3	Isolate_2	ᡂ	vrvgsilga_or_chachilga	U+1868
4	U+1820	A	A-4	Initial_0	ᠵ+ᠸ	titem + achvg	U+1865 + U+186A
5	U+1820 + FVS1	A	A-5	Initial_1	ᠵ	titem	U+1865
6	U+1820	A	A-6	Medial_0	ᠸ	achvg	U+186A
7	U+1820 + FVS1	A	A-7	Medial_1	ᠸ+ᠸ	achvg + achvg (used for 2nd half personal / mountain names, ᠥᠯᠭᠡᠢ᠋, ᠬᠠᠭᠠᠢ᠋)	U+186A + U+186A
8	U+ 1820 +FVS2	A	A-8	Final_0	᠋	orhich	U+1867
9	U + 1820 + FVS1	A	A-9	Final_1	ᠸ+ᡂ	achvg + vrvgsilga_or_chachilga	U+186A + U+1868
10	U+1820	A	A-10	Final_2	ᡂ	vrvgsilga_or_chachilga	U+1868
— Vowels（元音）- Letter E							
11	1821 + FVS2	E	E-1	Isolate_0	ᠶ+ᡂ	titem + vrvgsilga_or_chachilga	U+1865 + U+1868
12	1821 + FVS1	E	E-1	Isolate_1	ᡂ	vrvgsilga_or_chachilga	U+1868
13	1821 + NIRUGU	E	E-2	Initial_0	ᠶ	titem	U+1865
14	1821 + FVS1 + NIRUGU	E	E-3	Initial_1	ᠶ+ᠸ	titem + achvg	U+1865 + U+186A
15	U+1821	E	E-4	Medial_0	ᠸ	achvg	U+186A
16	U + 1821 +FVS2	E	E-6	Final_0	᠋	orhich	U+1867
17	U + 1821 + FVS	E	E-7	Final_1	ᡂ	vrvgsilga_or_chachilga	U+1868

18	MVS + U+1821	E	E-8	Final_2	ᠵ	vrvgsilga_or_chachilga	U+1868
— Vowels（元音）- Letter I							
19	U+1822 +FVS3	I	I-1	Isolate_0	ᠠ + ᠑	titem + hvms	U+1865 + U+186B
20	U+1822 +FVS1	I	I-2	Isolate_1	᠑	hvms	U+186B
21	U+ 1822 +FVS2	I	I-3	Isolate_2	Nil	Not exist in Traditional Mongolian Scrip	Nil
22	U+1822 + ［FVS2］ + NIRUGU	I	I-4	Initial_0	ᠠ + ᠊	titem + silbi	U+1865 + U+186C
23	U+1822 + FVS1 + NIRUGU	I	I-5	Initial_1	᠊	silbi	U+186C
24	NIRUGU + 1822 + FVS3 + NIRUGU	I	I-6	Medial_0	᠊	silbi	U+186C
25	NIRUGU + 1822 + FVS2 + NIRUGU	I	I-7	Medial_1	᠊ + ᠊	silbi + silbi	U+186C + U+186C
26	NIRUGU + 1822 + FVS1 + NIRUGU	I	I-8	Medial_2	᠋ + ᠊	achvg + silbi	U+186A + U+186C
27	NIRUGU + U+1822	I	I-9	Final_0	᠑	hvms	U+186B
28	U+1822 (this scenario does not exist in Unicode standard, it was covered in Opentype by Font.)	I	I-10	Final_1	᠊ + ᠈	silbi + Odoi_suul (ᠪᠠ = ᠪ + ᠊ + ᠈)	U+186C + U+1866
— Vowels（元音）- Letter O/U • Unification – The vowels O and U are unified into a single letter group, as their written forms are same in Traditional Mongolian script. Unicode encoding represents orthographic form rather than phonetic variation, and therefore does not distinguish between different vowel sounds when the glyph structure is the same.							
29/ 30	U+1823 / U+1824	O/U	O-1	Isolate_0	ᠠ + ᠊	titem + gedes	U+1865 + U+186D
31/ 32	U+1823 + FVS1	O/U	O-2	Isolate_1	ᠪ	b_banbvr_baga	U+1873

	/ U+1824 + FVS1						
33/ 34	U+1823 + FVS1 / U+1824 + FVS2	O/U	O-3	Isolate_2	Nil	Not exist in Traditional Mongolian Scrip	Nil
35/ 36	U+1823 / U+1824	O/U	O-2	Initial_0	ᠲ + ᠸ	titem + gedes	U+1865 + U+186D
37/ 38	U+1823 / U+1824	O/U	O-2	Initial_1	ᠸ	gedes	U+186D
39/ 40	U+1823 / U+1824	O/U	O-3	Medial_0	ᠸ	gedes	U+186D
41/ 42	U+1823 / U+1824	O/U	O-4	Medial_1	ᠷ + ᠸ	achvg + gedes	U+186A + U+186D
43/ 44	U+1823 / U+1824	O/U	O-5	Final_0	ᠪ	b_banbvr_baga	U+1873
45/ 45	U+1823 / U+1824	O/U	O-6	Final_1	ᠸ	gedes	U+186D
— Vowels（元音）- Letter OE/UE • Unification – The vowels OE and UE are unified into a single letter group, as their written forms are same in Traditional Mongolian script. Unicode encoding represents orthographic form rather than phonetic variation, and therefore does not distinguish between different vowel sounds when the glyph structure is the same.							
46/ 47	U+1825 / U+1826	OE/UE	U-1	Isolate_0	ᠲ + ᠸ + ᠂	titem + gedes + odoi_suul	U+1865 + U+186D + U+1866
48/ 49	U+1825 + FVS1 / U+1826 + FVS1	OE/UE	U-2	Isolate_1	ᠲ + ᠪ	titem + b_banbvr_baga	U+1865 + U+1873
50/ 51	U+1825 + FVS2 / U+1826 + FVS2	OE/UE	U-3	Isolate_2	ᠪ	b_banbvr_baga	U+1873
52/ 53	U+1825 + FVS3 / U+1826 + FVS3	OE/UE	U-4	Isolate_3	Nil	Not exist in Traditional Mongolian Scrip	Nil
54/ 55	U+1825 + FVS2 + NIRUGU /	OE/UE	U-5	Initial_0	ᠲ + ᠸ + ᠍	titem + gedes + silbi	U+1865 + U+186D + U+186C

	U+1826 + FVS2 + NIRUGU						
56/ 57	U+1825 + FVS1 + NIRUGU / U+1826 + FVS1 + NIRUGU	OE/UE	U-6	Initial_1	ᠳ	gedes	U+186D
58/ 59	U+1825 / U+1826	OE/UE	U-7	Medial_0	ᠳ	gedes	U+186D
60/ 61	U+1825 / U+1826	OE/UE	U-8	Medial_1	ᠳ＋᠍	gedes + silbi	U+186D + U+186C
62/ 63	U+1825 / U+1826	OE/UE	U-9	Medial_2	ᠬ＋ᠳ＋᠍	achvg + gedes + silbi	U+186A + U+186D + U+186C
64/ 65	U+1825 / U+1826	OE/UE	U-6	Final_0	ᠪ	b_banbvr_baga	U+1873
66/ 67	U+1825 / U+1826	OE/UE	U-7	Final_1	ᠳ	gedes	U+186D
68/ 69	U+1825 / U+1826	OE/UE	U-8	Final_2	ᠳ＋᠂	gedes + Odoi_suul	U+186D + U+1866
Vowels EE / Consonant WA — Unified Letter Group (EE/WA) • Vowel EE (a relatively recent addition, introduced primarily after the 1990s and likely influenced by the Cyrillic letter E (etc.. Европа) through Russian linguistic contact through Mongolia) and the consonant WA are unified into a single letter group. This is because their written forms are identical in Traditional Mongolian script. In accordance with Unicode principles, which encode orthographic form rather than phonetic distinction, no separate characters are assigned when multiple sounds share the same glyph structure.							
70/ 71	U+1827 / U+1838	EE / WA	W-1	Isolate	ᠴ	w_gohtai_silbi	U+189F
72/ 73	U+1827 / U+1838	EE / WA	W-2	Initial_0	ᠴ	w_gohtai_silbi	U+189F
74/ 75	U+1827 / U+1838	EE / WA	W-3	Medial_0	ᠴ	w_gohtai_silbi	U+189F
76/ 77	U+1827 / U+1838	EE / WA	W-4	Final_0	ᠴ＋᠂	w_gohtai_silbi + Odoi_suul	U+189F + U+1866
— Nasals（鼻音）							
78	U+1828	N	N-1	Isolate_0	ᠨ	n_titem	U+186F
79	U+1828 + FVS1	N	N-2	Isolate_1	ᠶ	titem	U+1865
80	U+1828	N	N-3	Initial_0	ᠨ	n_titem	U+186F
81	U+1828 + FVS1	N	N-4	Initial_1	ᠶ	titem	U+1865
82	U+1828	N	N-5	Medial_0	ᠨ	n_achvg	U+1871

83	U+1828	N	N-6	Medial_1	᠊	achvg Epenthetic letter/ 垫字（ᠠᠪᠬᠤᠷᠭᠠ ᠲᠡᠮᠳᠡᠭ）	U+186A
84	U+1828	N	N-7	Final_0	ᠨ᠋	n_orhich	U+1870
85	U+1828 + FVS2	N	N-8	Final_1	ᠨ	orhich Epenthetic letter/ 垫字（ᠠᠪᠬᠤᠷᠭᠠ ᠲᠡᠮᠳᠡᠭ）	U+1867
86	U+1829	NG	NG-1	Isolate	᠊ + ᠅	achvg + hinhir	U+186A + U+188A
87	U+1829	NG	NG-2	Initial_0	NIL	NIL	NIL
88	U+1829	NG	NG-3	Medial_0	᠊ + ᠅	achvg + hinhir	U+186A + U+188A
89	U+1829	NG	NG-4	Final_0	᠊ + ᠋	silbi + vrvgsilga_or_chachilga	U+186C + U+1868
90	U+1829	NG	NG-5	Final_1	᠊ + ᠊ + ᠋	achvg + silbi + vrvgsilga_or_chachilga	U+186A + U+186C + U+1868
— Core Consonants（基础辅音）							
91	U+182A	B	B-1	Isolate	ᠪ	b_banbvr_baga	U+1873
92	U+182A	B	B-2	Initial_0	ᠪ	b_banbvr_baga	U+1873
93	U+182A	B	B-3	Initial_1	ᠫ	b_banbvr_ih (to + ᠊ gedes)	U+1874
94	U+182A	B	B-4	Medial_0	ᠪ	b_banbvr_baga	U+1873
95	U+182A	B	B-5	Medial_1	ᠫ	b_banbvr_ih (to + ᠊ gedes)	U+1874
96	U+182A	B	B-6	Final_0	ᠬ	b_banbvr_egsilge	U+1872
97	U+182B	P	P-1	Isolate	ᠩ	p_chihi_baga	U+1876
98	U+182B	P	P-2	Initial_0	ᠩ	p_chihi_baga	U+1876
99	U+182B	P	P-3	Initial_1	ᠭ	p_chihi_ih (to + ᠊ gedes)	U+1877
100	U+182B	P	P-4	Medial_0	ᠩ	p_chihi_baga	U+1876
101	U+182B	P	P-5	Medial_1	ᠭ	p_chihi_ih (to + ᠊ gedes)	U+1877
102	U+182B	P	P-6	Final_0	ᠭ	p_chihi_egsilge	U+1875
103	U+182C	H	H-1	Isolate	ᠰ	h_hanhar	U+1878
104	U+182C	H	H-2	Initial_0	ᠰ	h_hanhar	U+1878
105	U+182C	H-HE	H-3	Initial_1	᠅	hinhir	U+188A
106	U+182C	H-HU	H-4	Initial_2	᠇	hunhur (to + ᠊ gedes)	U+188B
107	U+182C	HA	H-5	Medial_0	᠊ + ᠊	achvg + achvg	U+186A + U+186A
108	U+182C	H-HE	H-6	Medial_1	᠅	hinhir	U+188A

109	U+182C	H-HU	H-7	Medial_2	ᠴ	hunhur (to + ᠣ gedes)	U+188B
110	U+182C	H	H-8	Final_0	ᡴ	agsilga	U+1887
111	U+182D	G	GE-1	Isolate	ᡤ	g_hanhar	U+1888
112	U+182D	G	GE-2	Origin_1	ᠭ + ᠭ	achvg + achvg (ancient usage)	U+186A + U+186A
113	U+182D	G	GE-3	Initial_0	ᡤ	g_hanhar	U+1888
114	U+182D	G-GE	GE-4	Initial_1	ᠶ	hinhir	U+188A
115	U+182D	G-GU	GE-5	Initial_2	ᠴ	hunhur (to + ᠣ gedes)	U+188B
116	U+182D	G	GE-6	Medial_0	ᠻ + ᠻ	n_achvg + n_achvg	U+1871 + U+1871
117	U+182D	G-GE	GE-7	Medial_1	ᠶ	hinhir	U+188A
118	U+182D	G-GU	GE-8	Medial_2	ᠴ	hunhur (to + ᠣ gedes)	U+188B
119	U+182D	G	GE-9	Final_0	ᡴ	agsilga	U+1887
120	U+182D	G	GE-10	Final_1	ᡵ	dvsaltai_agsilga	U+1889
121	U+182D	G	GE-11	Final_2	ᠷ + ᠵ	silbi + vrvgsilga_or_chachilga	U+186C + U+1868
122	U+182E	M	M-1	Isolate	ᡨ	m_titem	U+188C
123	U+182E	M	M-2	Initial_0	ᡨ	m_titem	U+188C
124	U+182E	M	M-3	Medial_0	ᡮ	m_achvg	U+188E
125	U+182E	M	M-4	Final_0	ᡶ	m_hurbelge	U+188D
126	U+182F	L	L-1	Isolate	ᡩ	l_titem	U+188F
127	U+182F	L	L-2	Initial_0	ᡩ	l_titem	U+188F
128	U+182F	L	L-3	Medial_0	ᡯ	l_achvg	U+1891
129	U+182F	L	L-4	Final_0	ᡰ	l_hurbelge	U+1890
130	U+1830	S	S-1	Isolate	ᠰ	s_sechig	U+1894
131	U+1830	S	S-2	Initial_0	ᠰ	s_sechig	U+1894
132	U+1830	S	S-3	Medial_0	ᠰ	s_sechig	U+1894
133	U+1830	S	S-4	Final_0	ᡧ	s_agshilga	U+1893
134	U+1831	SH	SH-1	Isolate	ᡱ	x_sechig	U+1897
135	U+1831	SH	SH-2	Initial_0	ᡱ	x_sechig	U+1897
136	U+1831	SH	SH-3	Medial_0	ᡱ	x_sechig	U+1897
137	U+1831	SH	SH-4	Final_0	ᡲ	x_agshilga	U+1896
138	U+1832	T	T-1	Isolate	ᠲ	t_senj	U+1898
139	U+1832	T	T-2	Initial_0	ᠲ	t_senj	U+1898
140	U+1832	T	T-3	Medial_0	ᠳ	d_senj (swapped with D in history)	U+1899
141	U+1832	T	T-4	Final_0	ᠲ + ᠂	t_senj + Odoi_suul	U+1898 + U+1866
142	U+1833	D	D-1	Isolate	ᠳ	d_senj	U+1899
143	U+1833	D	D-2	Initial_0	ᠳ	d_senj	U+1899
144	U+1833	D	D-3	Medial_0	ᠲ	t_senj	U+1898

						(swapped with T in history)	
145	U+1833	D	D-4	Medial_1	+	gedes + achvg Epenthetic letter/ 垫字（[illegible]）	U+186D + U+186A
146	U+1833	D	D-5	Final_0	+	d_senj + Odoi_suul	U+1899 + U+1866
147	U+1833	D	D-6	Final_1	+	gedes + orhich Epenthetic letter/ 垫字（[illegible]）	U+186D + U+1867
148	U+1834	CH	CH-1	Isolate	ᠴ	q_acha	U+189A
149	U+1834	CH	CH-2	Initial_0	ᠴ	q_acha	U+189A
150	U+1834	CH	CH-3	Medial_0	ᠴ	q_acha	U+189A
151	U+1834	CH	CH-4	Final_0	+	q_acha + Odoi_suul	U+189A + U+1866
152 153	U+1835	J	J-1	Isolate	ᠵ	j_bosoo_silbi	U+189B
154 155	U+1835	J	J-2	Initial_0	ᠵ	j_bosoo_silbi	U+189B
156 157	U+1835	J	J-3	Medial_0	ᠵ	j_bosoo_silbi	U+189B
158 159	U+1835	J	J-4	Final_0	+	j_bosoo_silbi + Odoi_suul	U+189B + U+1866
160 161	U+1836	Y	Y-1	Isolate	ᠶ	y_erteger_silbi	U+189C
162	U+1836	Y	Y-2	Initial_0	ᠶ	y_erteger_silbi	U+189C
163	U+1836	Y	Y-3	Medial_0	ᠶ	y_erteger_silbi	U+189C
164	U+1836	Y	Y-4	Final_0	ᠶ	hvms	U+186B
165	U+1837	R	R-1	Isolate	ᠷ	r_erbgeljin_silbi	U+189E
166	U+1837	R	R-2	Initial_0	ᠷ	r_erbgeljin_silbi	U+189E
167	U+1837	R	R-3	Medial_0	ᠷ	r_erbgeljin_silbi	U+189E
168	U+1837	R	R-4	Final_0	ᠷ	r_Odoi_suul	U+189D

— Loan / Extended（借词/扩展）

.

- **Unification of K Letters** - The current Unicode standard encodes two separate characters at U+183A and U+183B, representing two forms of the letter K, commonly interpreted as regional variants (e.g., Mongolia and Inner Mongolia). This separation is not based on orthographic necessity, as both characters represent the same underlying letter and share the same structural role in Traditional Mongolian script. Accordingly, this model unifies the two K characters into a single letter group, with positional and stylistic differences handled at the variant level rather than through separate code points.

169	U+1839	F	F-1	Isolate	ᠹ	f_chihi_baga	U+18A1
170	U+1839	F	F-2	Initial_0	ᠹ	f_chihi_baga	U+18A1
171	U+1839	F	F-3	Initial_1	ᠹ	f_chihi_ih	U+18A2

						(to + ᠊ gedes)	
172	U+1839	F	F-4	Medial_0	ᠹ	f_chihi_baga	U+18A1
173	U+1839	F	F-5	Medial_1	ᠹ	f_chihi_ih (to + ᠊ gedes)	U+18A2
174	U+1839	F	F-6	Final_0	ᠽ	f_chihi_egsilge	U+18A0
175/ 176	U+183A/ U+183B	K/KH	K-1	Isolate	ᠺ	k_baga	U+18A4
177/ 178	U+183A/ U+183B	K/KH	K-2	Initial_0	ᠺ	k_baga	U+18A4
179/ 180	U+183A/ U+183B	K/KH	K-3	Initial_1	ᠻ	k_ih (to + ᠊ gedes)	U+18A5
181/ 182	U+183A/ U+183B	K/KH	K-4	Medial_0	ᠺ	k_baga	U+18A4
183/ 184	U+183A/ U+183B	K/KH	K-5	Medial_1	ᠻ	k_ih (to + ᠊ gedes)	U+18A5
185/ 186	U+183A/ U+183B	K/KH	K-6	Final_0	ᠾ	k_egsilge	U+18A3
187	U+183C	TS	TS-1	Isolate	ᠼ	c_acha	U+18A6
188	U+183C	TS	TS-2	Initial_0	ᠼ	c_acha	U+18A6
189	U+183C	TS	TS-3	Medial_0	ᠼ	c_acha	U+18A6
190	U+183C	TS	TS-4	Final_0	ᠼ + ᠂	c_acha + Odoi_suul	U+18A6 + U+1866
191	U+183D	Z	Z-1	Isolate	ᠽ	z_acha	U+18A7
192	U+183D	Z	Z-2	Initial_0	ᠽ	z_acha	U+18A7
193	U+183D	Z	Z-3	Medial_0	ᠽ	z_acha	U+18A7
194	U+183D	Z	Z-4	Final_0	ᠽ + ᠂	z_acha + Odoi_suul	U+18A7 + U+1866
— **Galig**（Extensions for Sanskrit and Tibetan 梵文/藏文转写扩展）							
	U+183E	HAA	HAA-1	Isolate	ᠲ + ᠷ	titem + zhi (galig)	U+1865 + U+18A8
	U+183F	ZRA	ZRA-1	Isolate	ᠿ	galig_tvvrai	U+18AA
	U+1840	LHA	LHA-1	Isolate	ᡀ	l_titem + zhi (galig)	U+188F + U+18A8
— **Galig**（Extensions for Chinese 汉文/中文转写扩展）							
	U+1841	ZHI	ZHI-1	Isolate	ᠷ	zhi (galig)	U+18A8
	U+1842	CHI	CHI-1	Isolate	ᠶ	gedes + gedes	U+186D + U+186D

Based on statistical analysis from the legacy Munk_Gal font base, Traditional Mongolian script is often described as having 236 distinct letter variants.

However, during our independent review and implementation, we found that a significant number of these variants are duplicated in structure and function. After normalization and deduplication, we identified 176 practically distinct variants that are sufficient for real-world usage of Traditional Mongolian.

This count does not include variants specific to Todo, Manchu, or Sibo scripts. These scripts can be added modularly without affecting the core architecture.

The ZCode component set functions as a collection of atomic, LEGO-like building blocks. These components can be freely combined without embedding grammatical logic at the glyph level. Any language-specific grammar rules, contextual behaviors, or special usages are instead handled at the Input Method Editor (IME) layer, where required components and variants are selected dynamically.

This separation of glyph composition and language logic ensures flexibility, extensibility, and long-term maintainability.

7.4 General Encoding Principles

The following principles govern the proposed encoding model:

1. **Elements are Unicode characters**, not glyph variants.
2. **Letter forms are represented by explicit element sequences**, not by font-level substitution.
3. **Structural variation is encoded**, while stylistic variation remains a font concern.
4. **Equivalent written forms produce identical Unicode sequences**, ensuring data consistency.

This approach allows all initial, medial, and final forms to be represented deterministically at the Unicode level, eliminating dependence on font-specific shaping logic.

7.5 Advantages of the Element-Based Model

The proposed element-based encoding model provides the following benefits:

- Precise representation of written Mongolian forms
- Elimination of ambiguous base-letter encoding
- Platform-independent rendering behavior
- Reliable search, normalization, and AI processing
- Reduced reliance on OpenType GSUB/GPOS logic

7.6 Next Steps

Sample encoded text demonstrating real-world usage

[illegible]

Zcode breakdown:

[illegible]

[illegible]

Zcode breakdown:

[illegible]