L2/15-012

Proposal to add
U+2B95 RIGHTWARDS BLACK ARROW
to Unicode Emoji

J. S. Choi, 2015-12-12

Abstract

In the Unicode Standard 7.0 from 2014, ➕ U+2B95 was added with the intent to complete the family of black arrows encoded by ⬅⬆⬇ U+2B05–U+2B07.
However, due to historical timing, ➕ U+2B95 was not yet encoded when the Unicode Emoji were first encoded in 2009–2010, and thus the family of four emoji black arrows were mapped not only to ⬅⬆⬇ U+2B05–U+2B07 but also to ➡ U+27A1—a compatibility character for ITC Zapf Dingbats—instead of ➕ U+2B95.
It is thus proposed that ➕ U+2B95 be added to the set of Unicode emoji characters and be given emoji- and text-style standardized variants, in order to match the properties of its siblings ⬅⬆⬇ U+2B05–U+2B07, with which it is explicitly unified.

Introduction

This document primarily discusses five encoded characters, already in Unicode as of 2015:

➡ **U+2B95 RIGHTWARDS BLACK ARROW:**
The main encoded character being discussed.
Located in the Miscellaneous Symbols and Arrows block.

⬅⬆⬇ **U+2B05–U+2B07 LEFTWARDS, UPWARDS, and DOWNWARDS BLACK ARROW:**
The three black arrows that ➡ U+2B95 completes.
Also located in the Miscellaneous Symbols and Arrows block.

➡ **U+27A1 BLACK RIGHTWARDS ARROW:**
A compatibility character for ITC Zapf Dingbats.
Located in the Dingbats block.

This document proposes the addition of ➡ U+2B95 to the set of emoji characters as defined by Unicode Technical Report (UTR) #51: "Unicode Emoji". In other words, it proposes:

1. A property change: ➡ U+2B95 should be given the EMOJI property defined in UTR #51.
2. Two new standardized variation sequences:
   ➡ U+2B95 followed by U+FE0E VARIATION SELECTOR-15 and
   ➡ U+2B95 followed by U+FE0F VARIATION SELECTOR-16.

The proposal first reviews the **history of the arrows' encoding** as background, then discusses ➡ **U+2B95 as emoji** in terms of the **Unicode Emoji selection factors**.

Some arrows are colored red for emphasis: ➡.
Red is not intended to convey any meaning other than emphasis.

History of the arrows' encoding

1993: ITC Zapf Dingbats

In **1993**: The glyphs from an important typeface, ITC Zapf Dingbats series 100 by the late Hermann Zapf, were encoded in the Unicode Standard (UCS) 1.1 for compatibility with PostScript printers that used them. This included ➡ U+27A1 BLACK RIGHTWARDS ARROW.

The Zapf Dingbats arrows all face rightwards, as generically rotatable arrow glyphs. No leftwards, upwards, or downwards versions of arrows were encoded because PostScript printers were assumed to rotate generic rightwards arrows in original Zapf Dingbats fonts. ➡ U+27A1's representative glyph was thus taken from Zapf Dingbats. [Whistler 2015]

2003: DPRK KPS 9566

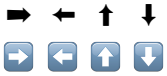
In **2003**: Representatives of the the Committee of Standardization of the DPRK (North Korea) submitted a proposal, ISO WG2 N2374: "Proposal to Add of 70 Symbols to ISO/IEC 10646-1:2000". The proposal requested the encoding of compatibility characters for the DPRK encoding standard KPS 9566. These compatibility characters included black-filled arrows in the four cardinal directions.

The DPRK proposal only included leftwards, upwards, and downwards black arrows, apparently because the representatives believed that ➡ U+27A1 fit their purposes for compatibility with their rightwards black arrow.

The former three were encoded as ⬅⬆⬇ U+2B05–U+2B07 in the UCS 4.0. Their representative glyphs and names were taken directly from the DPRK proposal; the glyphs' widths and names' orderings thus did not align with ➡ U+27A1 (e.g., ⬅ LEFTWARDS BLACK ARROW vs. ➡ BLACK RIGHTWARDS ARROW), but no comments about the three characters were apparently submitted. They were not mapped to any commercial sets at the time. [Whistler 2002; Whistler 2015]

Unification of new KPS-9566 compatibility arrows ⬅⬆⬇ U+2B05–U+2B07 with rotations of Zapf Dingbat ➡ U+27A1 was implied by the UCS but was not explicit. For the next decade, most fonts implementing all four characters used glyphs matching those in the code charts (i.e., the mismatching Zapf Dingbat glyph for the right arrow, and the KPS-9566 glyphs for the other black arrows). [Suignard 2015; Whistler 2015]

2009–2010: ARIB and Japanese cellular carriers; Unicode Emoji

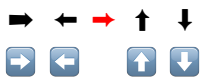


In **2009–2010**: As documented by UTR #51, major software vendors Google, Apple, and Microsoft began to support emoji characters from the Association of Radio Industries and Businesses of Japan (ARIB) and various Japanese cellular carriers using characters from the UCS 6.0. Four of those Japanese-carrier characters were black arrows in the four cardinal directions.

The three companies mapped three of the black arrows to the KPS-9566-compatibility black arrows ⬅⬆⬇ U+2B05–U+2B07. However, presumably because it was assumed to be part of their family and there was no better alternative, the Zapf Dingbat ➡ U+27A1 for the final, rightwards black arrow from the ARIB/Japanese-carrier emoji.

Based on then-current usage, these four characters' mappings and other properties were added to files of Unicode Emoji data, where they remain to this day. Today, these data have not yet been formally defined by the UCS and are awaiting further discussion.

2011–2014: Wingdings and Webdings; total arrow re-rationalization



In **2014**: An extensive re-rationalization of all arrows symbols occurred in the UCS 7.0, due to the addition of arrows from Wingdings and Webdings [Whistler 2015].

ISO WG2 N4143, the proposal for adding Wingdings/Webdings to Unicode, included two parallel sets of arrows, ten white and ten black:
⇦⇨⇧⇩⇗⇖⇘⇙⇔⇕➡➡⬆⬇⬈⬉⬊⬋⬌⬍.

N4143 unified all but one of these Wingding/Webding white/black arrows with already-encoded characters. The encoded characters had differing and mismatching representative glyphs in the UCS at the time, and N4143 thus proposed replacing their then-existing representative glyphs with Wingding's harmonized glyphs. (This proposal uses similarly harmonized glyphs throughout.) These unifications were:

⇦⇧⇨⇩	with arrows U+21E6–U+21E9 from the UCS 2.0,
⇕	with arrow U+21F3 from the UCS 3.0,
⇗⇖⇘⇙⇔	with KPS-9566-compatibility arrows U+2B00–U+2B04,
➡⬆⬇	with KPS-9566-compatibility arrows U+2B05–U+2B07,
⬈⬉⬊⬋⬌⬍	with KPS-9566-compatibility arrows U+2B08–U+2B0D, and
➡	with **no** already-encoded character.

One arrow, the rightwards black arrow ➡, was unified with **no** already-encoded character; it had been determined that no then-existing encoded characters were appropriate for it. In particular, ➡ was not unified with the Zapf Dingbat ➡ U+27A1. Indeed, N4143 disunified the Zapf Dingbats from the Wingdings in general because "their shapes are typically different".

After deliberation on N4143, an ad-hoc group from the ISO JTC1/SC2/WG2 decided thus:

> …The document N4143 asks for reunification of 10 characters (Wingdings-239 to 248, corresponding to the white arrows [⇦⇨⇧⇩⇗⇖⇘⇙⇔⇕])…[which is] associated with a request to use the glyphs from the Wingdings set to represent these 10 characters in the UCS. The 10 characters in question are consistent among themselves and are also more compatible in design with keyboard symbols which are typically associated with these symbols…

> Checking on how some commercial fonts implemented these characters (mostly Japanese and symbol oriented ones) shows that they vary widely, not implementing the current 'skinny' type but mostly varying on fat shapes…
>
> …the argument for preserving current [representative] shapes [in the UCS]…is not that strong. The Wingdings shapes are much closer to what is currently implemented [by commercial fonts] and they are also closer to what would be expected for the representation of related keyboard symbols.
>
> The glyph change for these 10 characters [⇦⇨⇧⇩⇗⇖⇘⇙⇔⇳] (21E6..21E9, 21F3, 2B00..2B04) is accepted…**Finally, the glyphs for 9 black arrows [⬅⬆⬇⬈⬉⬊⬋⬌⬍] in the range 2B05..2B0D are also updated.**
>
> **To complete the set of BLACK ARROW in 2B05..2B0D a new character is added: [➡] 2B95 RIGHTWARDS BLACK ARROW (the character 27A1 BLACK RIGHTWARDS ARROW [➡] in the dingbat block is not an appropriate match for the other 9 characters).** [2012; emphasis added]

Thus in the UCS 7.0, the black arrows ⬅⬆⬇ U+2B05–U+2B07, originally for KPS-9566 compatibility, were unified with similar Wingding black arrows, and their representative glyphs are modified thus to harmonize with one another. However, the glyph of Zapf Dingbat arrow ➡ U+27A1 was deemed to be unmodifiable, and its identity was deemed to be strongly coupled to the original arrow glyph in the ITC Zapf Dingbat typefaces; its glyph remained that from the original typeface by Zapf.

The now-generic black arrows ⬅⬆⬇ U+2B05–U+2B07 were thus disunified from rotations of ➡ U+27A1. A new character, ➡ U+2B95 RIGHTWARDS BLACK ARROW, was added with the intention of completing the ⬅⬆⬇ U+2B05–U+2B07 family. The four of them were unified by the UCS and received newly drawn matching representative glyphs. [Suignard 2015; Whistler 2015]

Status quo

Thus **today**: The relatively new ➡ U+2B95 RIGHTWARDS BLACK ARROW is unified with the now-generic ⬅⬆⬇ U+2B05–U+2B07, together forming a single family of arrows. It will take time for ➡ U+2B95 to be implemented by new fonts, but the UCS is clear about their unification. ➡ U+27A1 has been clearly **disunified** from the family and is now merely a Zapf Dingbat. As font support for ➡ U+2B95 gradually improves, so too will its usage as a sibling of ⬅⬆⬇ U+2B05–U+2B07.

However, this is still not yet completely true: UTR #51 and the Unicode Emoji data files currently define the rightwards version of ⬅⬆⬇ U+2B05–U+2B07 to be the Zapf Dingbat ➡ U+27A1; they currently **exclude** U+2B95 from being an emoji character. Furthermore, there are no text/emoji standardized variants of U+2B95 yet, unlike ➡ U+27A1.

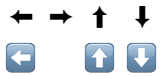
Upon reviewing the history above, it becomes apparent that this is due to missed timing between the advent of Unicode Emoji (in 2009–2010) and the advent of ➡ U+2B95 (in 2011–2014). Apple, Google, and Microsoft had no character other than ➡ U+27A1 that they could use for the Japanese carriers' rightwards black arrow at the time.

Emoji selection factors

Perhaps uniquely for Unicode Emoji proposals, this proposal's character, ➡ U+2B95, is an already encoded character, with the express intent of completing a family of other encoded characters that are also emoji characters. Nevertheless, the appropriateness of its inclusion is here discussed using the same selection factors as those of other emoji characters [Unicode Consortium 2015, "Submitting Emoji character Proposals"]. The basis of adding ➡ U+2B95 to the set of emoji characters is based primarily on compatibility and completeness, and those factors are discussed first.

Compatibility and completeness

Since the publication of the UCS 7.0, there has been a hole in the four unified cardinal black arrows of today:



The UCS has unified ➡ U+2B95 with the rightwards rotation of ⬅⬆⬇ U+2B05–U+2B07 since version 7.0; this is now a characteristic inherent to their identities. Text-input systems may thus present ⬅➡⬆⬇ U+2B05–U+2B07, U+2B95 as a single black-arrow family to users, and text-processing systems may output ➡ U+2B95 when rotating/flipping/processing text containing ⬅⬆⬇ U+2B05–U+2B07. This is perhaps particularly true in conversion between writing systems with differing directionality, as arrow characters do not reflect between writing directions.

Furthermore, emoji characters may appear in many contexts and environments: formal (e.g., word processing), general (e.g., plain web pages), or informal (e.g., chat messages) [UTF #51]. However, it is a corollary of the Unicode character-encoding model's definitions that:

> No matter the environment, emoji characters maintain their identities.

Just as all other encoded characters maintain their identities in different styles and environments no matter the particular glyph or variant chosen, so too do ⬅⬆⬇ U+2B05–U+2B07 maintain their identities as siblings of ➡ U+2B95 no matter the environment, regardless of whether they happen to be in text style or emoji style.

Users of ⬅➡⬆⬇ U+2B05–U+2B07, U+2B95 may thus be surprised when the former's behavior does not match that of the latter. ➡ U+2B95 has been unified with ⬅⬆⬇ U+2B05–U+2B07, and ⬅⬆⬇ U+2B05–U+2B07 are emoji characters, yet their final sibling ➡ U+2B95 is not an emoji character. ➡ U+2B95 also lacks the standardized variants for text and emoji styles that its siblings have.

Indeed, inserting the complete cardinal-direction black-arrow family, ⬅➡⬆⬇, into an environment that prefers emoji style would today result in the surprising ⬅️➡️⬆️⬇️. Using U+FE0F VARIATION SELECTOR-16 on the four arrows would not resolve the problem—in fact, conformant implementations are currently forbidden to change ➡ U+2B95's presentation based on variation selectors [UCS 8.0].

It would be logical to complete the family of black-arrow emoji with their proper, unified rightwards counterpart, just as they have been in normal text since UCS 7.0. It would also improve compatibility with any text systems that use ➡ U+2B95 as the reflected/rotated version of ⬅⬆⬇ U+2B05–U+2B07 in general, no matter the environment.

Furthermore, ➡ U+2B95 was encoded in UCS 7.0 with the intent of being mapped to corresponding the rightwards black arrow in the Wingdings and Webdings sets, which are considered by UTR #51 to be major emoji sources. Giving ➡ U+2B95 the EMOJI property would thus also improve compatibility with a major emoji source.

It is **not** proposed to remove the EMOJI property from the Zapf Dingbat ➡ U+27A1. All standardized variants are permanently stable; ➡ U+27A1 must continue to be an emoji character both to fulfill stability guarantees and for compatibility with current usage. However, this does not in turn preclude the encouragement of using ➡ U+2B95 instead of ➡ U+27A1, in the same general contexts that ➡ U+2B95's siblings would also be used in, that is, all contexts other than those that specifically refer to the Zapf Dingbat.

Expected usage level

Because this character strongly resembles another character that already has the EMOJI property, ➡ U+27A1, the expected usage level strongly depends on the decisions of individual software vendors (see "Current representability" section below). For instance, if software vendors decide to modify their emoji-input pallets such that they enter ➡ U+2B95 instead of ➡ U+27A1, then current ➡ U+2B95's usage as emoji would replace that of the current ➡ U+27A1.

Regardless of vendor decisions, however, ➡ U+2B95 has already been encoded in Unicode and unified with ⬅⬆⬇ U+2B05–U+2B07. As font support for ➡ U+2B95 gradually improves, so too will its usage as a sibling of ⬅⬆⬇ U+2B05–U+2B07, including in instances where ⬅⬆⬇ U+2B05–U+2B07 are also used as emoji.

Current representability

A factor for Unicode Emoji exclusion is whether its the concept already be represented by encoded emoji characters. The answer in this case happens to be yes: the black right arrow in emoji form is already representable by the Zapf Dingbat ➡ U+27A1. Due to the late timing of ➡ U+2B95's encoding, ➡ U+27A1 had been the only existent character that was at all appropriate for mapping of ARIB's and Japanese carriers' emoji-style rightwards black arrows.

The "duplication" of the rightwards black arrow in the set of emoji may be of concern. However, this apparent redundancy has already been permanently established in the UCS proper for general text through its very encoding. Redundancy between ➡ U+2B95 and ➡ U+27A1 as emoji is little different than their redundancy in normal, general text.

The UCS (versions 7.0 and 8.0) is clear: ➡ U+2B95 and ➡ U+27A1 have been **disunified**. This, the encoding of ➡ U+2B95, and its unification with ⬅⬆⬇ U+2B05–U+2B07 are faits accomplis. The "redundancy" was deemed to be tolerable in the interests of compatibility and completeness. This proposal would merely finish the job and complete the already-established unification.

Should this proposal be accepted, confusion by emoji users is a risk. However, this may be mitigated: vendors' emoji-input palettes may present only ➡ U+2B95, replacing ➡ U+27A1, with little disadvantage. The status quo's inconsistency in character properties may well cause more user confusion than the change.

Another concern may be that compatibility mappings with ARIB or Japanese-carrier sets already use ➡ U+27A1. However, such mappings are not normatively defined or guaranteed to be stable by the UCS. Furthermore, such mappings are not necessarily injective anyway: the Japanese-carrier zodiac emoji, for instance, may be mapped to either Greek zodiac symbols or animal pictographs (e.g., ♈ U+27A1 versus 🐏 U+1F40F). Text-processing systems may change their mapping of legacy emoji-style rightwards black arrows from ➡ U+27A1 to ➡ U+2B95, but legacy mappings may also continue using ➡ U+27A1 with little cost.

Open-endedness

The arguments for inclusion above may be similarly applied to any already-encoded character that: (1) is a strongly unified sibling of emoji characters but (2) is not yet an emoji character itself. The set of such characters is a closed, finite, and small one. To the author's knowledge, its only other members are those discussed in the "Out-of-scope issue" section below. However, even if the set were hypothetically to be large, the case for its members' inclusion would still be compelling on the bases of compatibility, completeness, and the nature of the Unicode character model itself.

Other factors

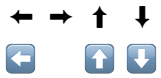
Due to the unique nature of this emoji proposal, image distinctiveness, frequent requesting, excessive specificity, and logo/brand/icon/signage/etc. are not applicable.

Character properties, glyphs, and collation

Under this proposal, all current properties of encoded character ➡ U+2B95 would stay the same save for one: it would be given the EMOJI property defined in UTR #51. It would **not** be given the EMOJI PRESENTATION or EMOJI MODIFIER BASE properties. Its emoji properties would afterwards match that of its siblings.

➡ U+2B95 would also be given standardized variation sequences matching those of its siblings ⬅⬆⬇ U+2B05–U+2B07; these would respectively use U+FE0E VARIATION SELECTOR-15 and U+FE0F VARIATION SELECTOR-16.

Its emoji style's glyphs would harmonize with its unified siblings'; in other words, it would share the same emoji glyphs as those of ➡ U+27A1.



It may also be prudent to add a note to the UCS informing the reader of the differences in intended usage between ➡ U+2B95 and ➡ U+27A1, whether in the core text or in the character-name list. The Unicode Emoji charts' black-and-white glyphs may also be updated to use the arrows' harmonized representative glyphs from UCS 7.0.

As for collation by the CLDR emoji ordering, it is proposed to place ➡ U+2B95 immediately before ➡ U+27A1, maintaining compatibility with legacy use of ➡ U+27A1 as the rightwards emoji arrow. User interfaces presenting emoji pallets for input would only show either ➡ U+2B95 or ➡ U+27A1 and not both.

Out-of-scope issue

A related issue involves the diagonal siblings of ➡ U+2B95 and ⬅⬆⬇ U+2B05–U+2B07: ↗↖↘↙↔↕ U+2B08–U+2B0D. These diagonal arrows were encoded in the first place as siblings of ⬅⬆⬇ U+2B05–U+2B07 in the same 2003 DPRK proposal. However, in 2009, the "skinny" arrows ↗↖↘↙ U+2196–U+2199 were used for the diagonal emoji arrows instead.

Mapping inconsistent sets of sibling encoded characters to emoji arrows suffers from the same compatibility and completeness issues described above for ➡ U+2B95: not only for the black arrows ⬅➡⬆⬇↗↖↘↙↔↕ but also for the "skinny" arrows ←→↑↓↗↖↘↙↔↕ U+2190–U+2199. A solution may to add both arrow families to the emoji characters, which would give users two families of emoji arrows with two widths, narrow and wide.

However, the author has chosen to make that other issue out of scope of this document's proposal, deferring it to after the consideration of ➡ U+2B95 as an emoji character.

Citations and acknowledgements

DPRK. Committee for Standardization.

"Proposal to Add of 70 Symbols to ISO/IEC 10646-1:2000". ISO JTC1/SC2/WG2 N2374. 2001-09-03. <http://unicode.org/L2/L2001/01349-N2374-DPRK-AddSymbols.pdf>.

Davis, Mark.

"Re: Emoji data in UCD xml ?". Message to unicode@unicode.org. 2015-10-29. <http://unicode.org/mail-arch/unicode-ml/y2015-m10/0226.html>.

Davis, Mark and Eberg, Peter.

"Unicode Technical Report #51: Unicode Emoji". Version 2.0. Unicode, Inc., 2015-11-12. <http://www.unicode.org/reports/tr51/tr51-5.html>.

ISO JTC1/SC2/WG2. Michael Suignard, ed.

"Disposition of comments on SC2 N4201". WG2 N4239. 2012-02-16. <http://unicode.org/L2/L2012/12130-N4239.pdf>.

Suignard, Michel.

"Re: Arrow dingbats". Message to unicode@unicode.org. 2015-05. <http://unicode.org/mail-arch/unicode-ml/y2015-m05/0268.html>.
(Note: This post contains paragraphs quoted from another person that is not marked differently, with Suignard's replies below each one.)

"Updated proposal to add Wingdings and Webdings Symbols". UTC L2/11-344; ISO JTC1/SC2/WG2 N4143. Unicode, Inc., 2011-09-23.

Unicode Consortium.

"Core Specification". *The Unicode Standard*, Version 8.0. Unicode, Inc., 2015.

"Dingbats". *The Unicode Standard*, Version 7.0. Unicode, Inc., 2014. <http://www.unicode.org/charts/PDF/Unicode-7.0/U70-2700.pdf>.
(Version 8.0 of this code chart was not available at time of writing, but no relevant changes occurred in this code block in version 8.0.)

"Miscellaneous Symbols and Arrows". *The Unicode Standard*, Version 8.0. Unicode, Inc., 2015. <http://www.unicode.org/charts/PDF/Unicode-8.0/U80-2B00.pdf>.

"Submitting Emoji character Proposals". *Unicode Emoji*. Unicode, Inc., 2015. <http://unicode.org/emoji/selection.html>.

Whistler, Ken.

"Early look at WG2 consent docket". UTC L2/02-292. Message to unicore@unicode.org. 2002-06-03. <http://www.unicode.org/L2/L2002/02292-whistler.txt>.

"Re: Arrow dingbats". Message to unicode@unicode.org. 2015-05-28. <http://unicode.org/mail-arch/unicode-ml/y2015-m05/0272.html>.

"Re: Emoji data in UCD xml ?". Message to unicode@unicode.org. 2015-10-29. <http://unicode.org/mail-arch/unicode-ml/y2015-m10/0223.html>.

This document uses and embeds the following fonts under license:

Adobe, Inc.

Minion Pro. 1990.

Douros, George.

Symbola 8.00. 2015-01-10. Informal free license. <http://users.teilar.gr/~g1951d>.