A colour font, drawn as text

The four marks below are not letters of any face and not drawings on this page. They are a COLOUR FONT: a face whose symbols are made of stacked outlines, each one pointing at an entry in a colour palette. PDF cannot embed such a face - the text operators paint an outline in one colour and nothing else - so tclpdf reads the layers and draws them into a Type 3 font, and the symbols become characters of a family this document carries.

From there on nothing knows the difference. They sit on the baseline, they take the font size, [textWidth] measures them, a paragraph breaks around them, and they come back out of pdftotext as the characters they stand for.

two layers, two layers, two layers, three - and the tick and the cross share one octagon glyph between them

Sequences: several characters, one glyph

The three marks below are not in the face's character map at all. Each is reached by a SEQUENCE - a rule in the face's ccmp feature that says which characters, in which order, are drawn as which glyph. That is how an emoji face makes a family out of three people and two joiners, a skin tone out of a hand and a modifier, and a flag out of two regional indicators; the mechanism is the same whatever the artwork is.

The caller does not take the string apart. -chars is TEXT, the package applies the face's own ccmp to it, and every glyph that comes out is one glyph of the font. Which characters belong together is a property of the face - not of Unicode, and not of whoever wrote the call.

warning + U+200D + cross, tick + modifier, indicator + indicator - three glyphs the character map cannot reach

measured at 22 pt: the warning alone is 7.761 mm wide, the three characters of the sequence are 7.761 mm - the same glyph, the same advance - while the two marks set side by side without the joiner are 15.522 mm, which is two glyphs and two advances.

A status column

The case this exists for. The mark is a character in the line, so the wording behind it starts where [textWidth] says and the row needs no arithmetic of its own.

- Invoice 2026-0412 sent, payment received
- Invoice 2026-0413 sent, payment received
- Invoice 2026-0414 sent, 14 days overdue
- Invoice 2026-0415 rejected by the recipient

It is a font

The same four marks at five sizes. Nothing is scaled by hand: the size is the font size, and the glyph streams in the file are the integers the face stores - the FontMatrix carries the em.

A layer in the colour of the text

The logo mark has three layers: a shadow at 25 % alpha, a left half in brand blue, and a right half whose palette index is the sentinel 0xFFFF - which is not an index but the instruction "take whatever colour the text has". So the right half follows -color and the left half does not. A single-colour font can do neither half; a bitmap can do neither.

In the running text

One string, one call, two faces: -fallback names the faces that may set what -family cannot, so the package decides per character where each glyph comes from and the paragraph breaks around the marks like any other character.

AND A SEQUENCE IS ONE UNIT IN THAT CHAIN. The withdrawn note near the end of each line is the three-character sequence from the page before, and no face has a glyph for the joiner in the middle of it - so the chain is asked for the LONGEST piece any of its faces can set, not for one character at a time. Without that the line would be refused where the joiner stands.

WHICH WAY ROUND MATTERS, and the two lines below are the same sentence set both ways. The chain is tried IN ORDER, and DejaVu Sans has a warning sign, a tick and a cross of its own - so with the words first the marks come out of DejaVu, in one colour, and only the logo reaches the colour font because nothing else has that character. With the colour font first they come out in colour and the letters fall through to DejaVu, which is what a face of symbols wants. Neither is a defect: a colour font is one face in a chain, and a chain has a direction.

⚠ Delivery 2026-0414 is overdue. The goods left the warehouse on the 3rd ✔ and were refused at the door on the 9th ✖ - please tell us what to do with them. 🔹 The delivery note was withdrawn ⚠ on the 11th.

the words first: the marks come from DejaVu Sans

⚠ Delivery 2026-0414 is overdue. The goods left the warehouse on the 3rd ✅ and were refused at the door on the 9th ❎ - please tell us what to do with them. 🔹 The delivery note was withdrawn ⚠ on the 11th.

the colour font first: the marks come from it

Why not just embed it

The same file handed to [font embed] is refused, and the refusal is the point: every character this face covers has an EMPTY outline, because all the drawing sits in the layer glyphs. Embedded, it would produce a document that is valid, extractable and blank, with nothing anywhere reporting it - not the API, not pdftotext, not pdffonts, not qpdf, not veraPDF.

The message, caught from this very page: tclpdf: "marks.ttf" draws nothing - every character it covers has an EMPTY outline. That is what a colour font looks like from the outside: its pictures sit in a table of its own (COLR/CPAL, CBDT, sbix, SVG) which this package does not write, and a document embedding it would come out blank with nothing reporting it. Embed a monochrome face instead - Noto Emoji for Noto Color Emoji - or draw the symbols with [font define] and [font glyph] as a Type 3 font

Check it yourself

The first command shows the font dictionary: no font file anywhere, a /CharProcs entry per symbol, and an /ExtGState in the font's own /Resources - the one the translucent shadow needs. The second prints the glyph stream of the logo: d0, then one q/Q bracket per layer, and the third bracket has no colour operator in it at all, which is the sentinel. The third command is the one that matters for a reader.

```
qpdf --qdf --object-streams=disable 02.14-colour-font.pdf - | grep -A20 Type3
qpdf --qdf --object-streams=disable 02.14-colour-font.pdf - | grep -B2 -A24 ' d0'
pdftotext 02.14-colour-font.pdf - | head -20
pdffonts 02.14-colour-font.pdf
```