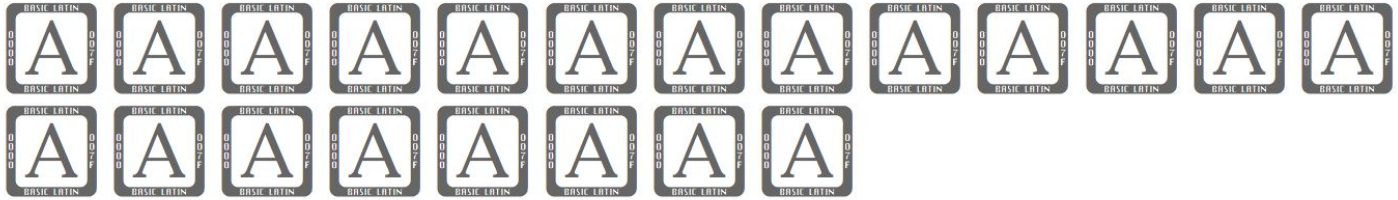
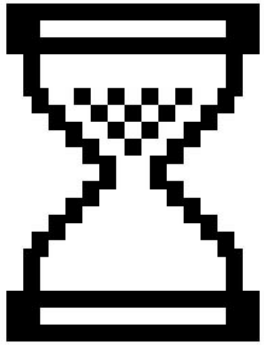

BASIC LATIN
A
0000
007F


BASIC LATIN
A
LATIN-1
Á
0000
007F


BASIC LATIN
A
0000
007F



With permission from William Löthman: https://dribbble.com/shots/2112159-Hourglass-Icon

النص وتخطيط نظرة عامة

BlinkOn 5 - Dominik Röttsches

نوفمبر 2015

Text & Layout Overview

BlinkOn 5 - Dominik RÃ¶ttsches

November 2015

Text & Layout Overview

BlinkOn 5 - Dominik RÃ¶ttsches

November 2015

Text & Layout Overview

BlinkOn 5 - Dominik Röttsches

November 2015

テキストとレイアウトの概要

Blinkon 5
- Dominik Röttsches

November 2015

Text & Layout Overview

BlinkOn 5 - Dominik Röttsches

November 2015

Text & Layout Overview

BlinkOn 5 - Dominik Röttsches (drott@)

November 2015

Agenda

- Goals & Challenges
- What does the Font code do?
- Recent Improvements
- Future Plans

Challenges of Text & Layout (Incomplete)

- Complexity & Dependencies
- Pixel precision
- Optimal Font Matching & Fallback
- Shaping non-Latin languages
- Vertical text
- Performance
- Typographic features

Goals of Text & Layout

- Correct and consistent display of all text in all languages
- High performance for complex text processing
- Low overhead & cost for advanced typographic features

What does *Font.h* do?

- Measuring Text
- Drawing Text
- Selecting Text

Measuring Text

Measuring Text

$$length = f(font, size, \quad text \quad)$$

Measuring Text

$$length = f\,(\,font,\ size,\ direction,\ features,\ script,\ language,\ text,\ context\,)$$

Measuring Text

length = f (font, size, direction, features, script, language, text, context)

Measuring Text

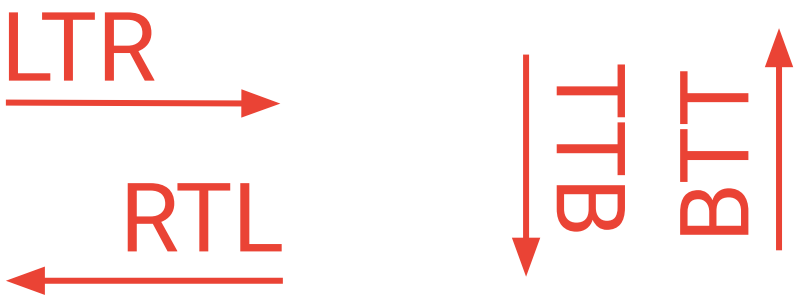
32

16

6

length = f (font, size, direction, features, script, language, text, context)

Measuring Text



length = f (font, size, direction, features, script, language, text, context)

Measuring Text

- Small Caps
- Ligatures
- Swashes, Ornamentals
- Number Forms
- … more

length = f (font, size, direction, features, script, language, text, context)

Measuring Text

- Latin
- Arabic
- Hiragana
- Cyrillic
- Traditional & Simplified Chinese
- “Common”
- … more

length = f (font, size, direction, features, script, language, text, context)

Measuring Text

Language	***Script***
English, German, Finnish…	*Latin*
Azerbaijani	*Cyrillic, Latin, Arabic*
Chinese	*Traditional, Simplified*

length = f (font, size, direction, features, script, language, text, context)

Measuring Text

LoremHello WorldIpsum

length = f (font, size, direction, features, script, language, text, context)

Measuring Text

length = f (font, size, direction, features, script, language, text, context)

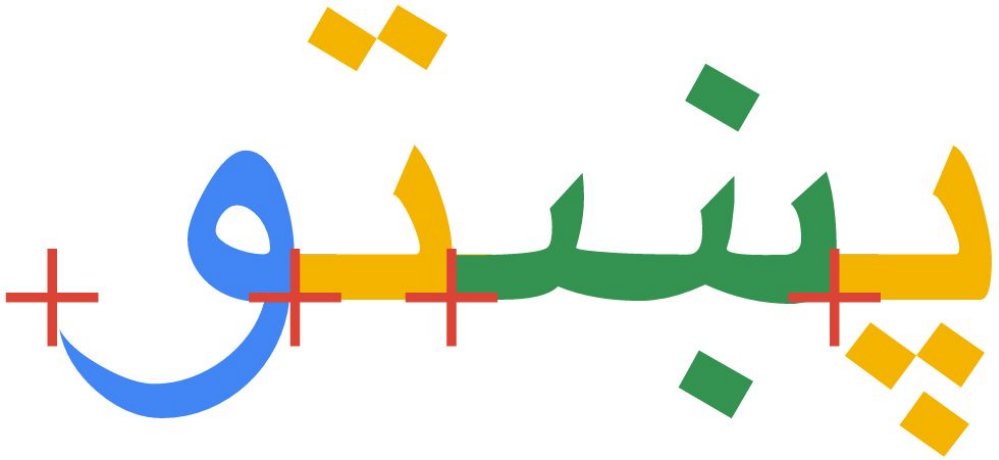
Measuring Text

پ بنت و

length = f (font, size, direction, features, script, language, text, context)

Drawing Text

Drawing Text

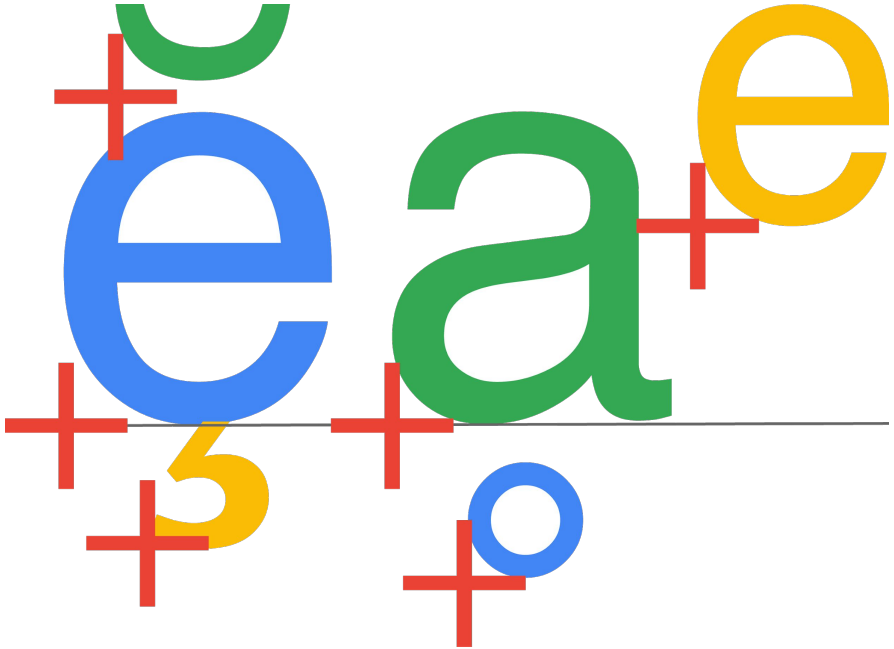


Drawing Text

Drawing Text

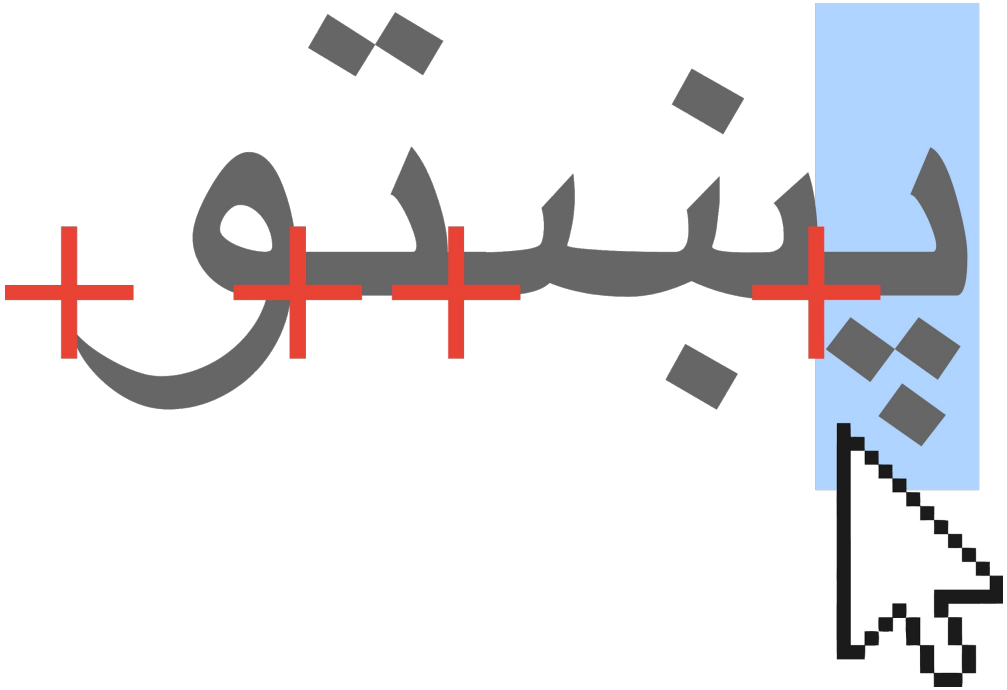
Drawing Text

`GlyphBuffer`



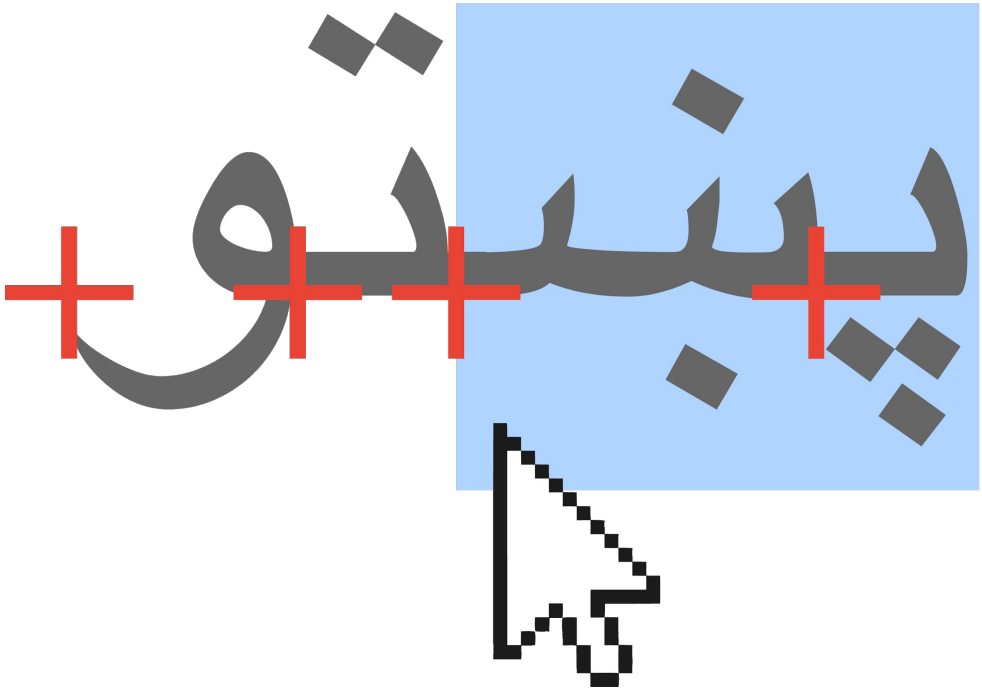
Selecting Text

Selecting Text



Character Position ⇄ Screen Coordinates

Selecting Text



Character Position ⇄ Screen Coordinates

Font.h does:

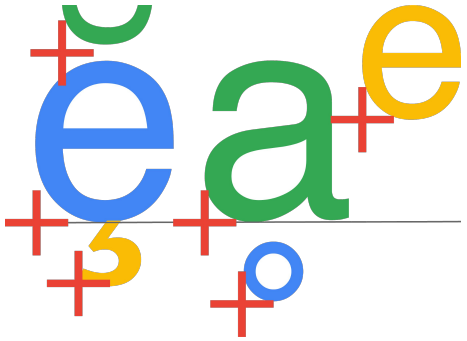
- Measuring Text
- Drawing Text
- Selecting Text

Shaping

Shaping

پ بنت و

پښتو



f (font, size, direction, features, script, language, text, context)

Simple vs Complex

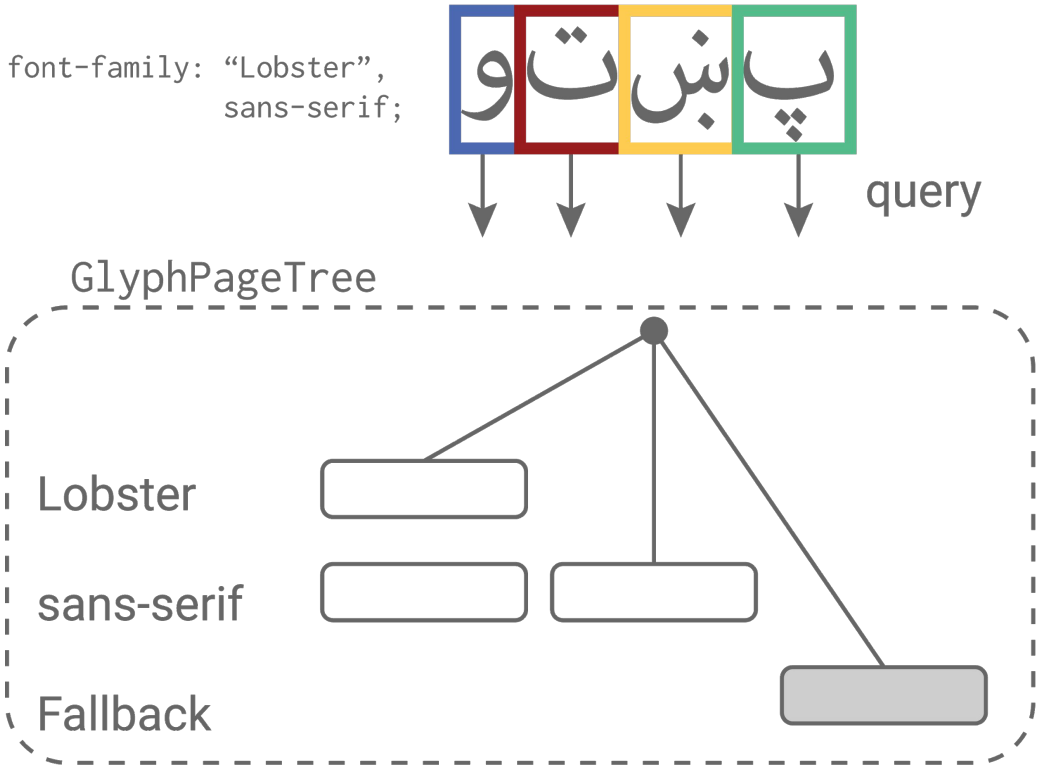
Simple vs. Complex Text Path

Simple

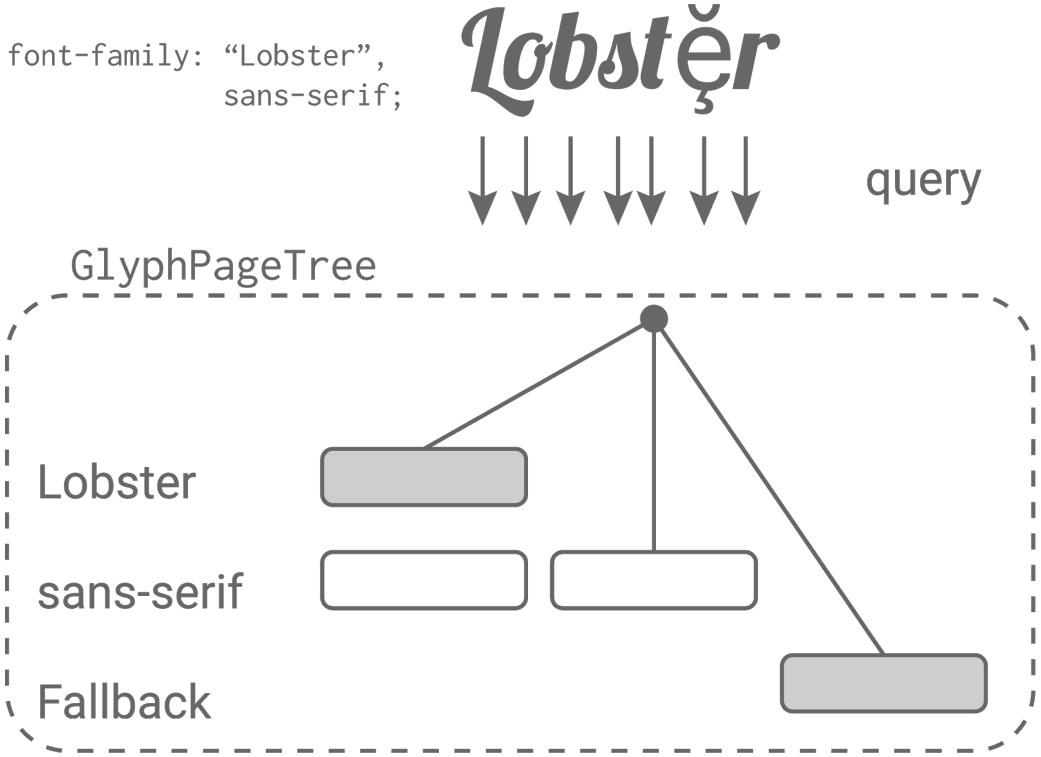
Complex


```
font-family: “Lobster”, sans-serif;
```

Simple Path


font-family: “Lobster”,
sans-serif;
پښتو
query
GlyphPageTree
Lobster
sans-serif
Fallback

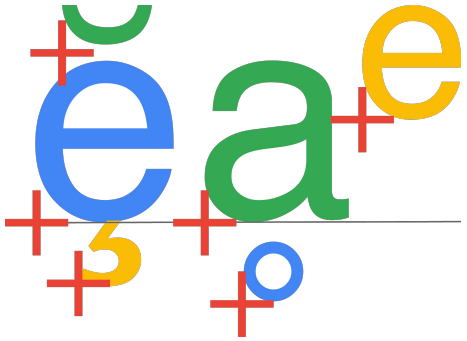
Simple Path


font-family: “Lobster”,
sans-serif;
Lobstęr
query
GlyphPageTree
Lobster
sans-serif
Fallback

Complex Path = Full Shaping

پ بنت و

پښتو



length = f (font, size, direction, features, script, language, text, context)

Simple vs. Complex Text Path

Simple Path	Complex Path
Typewriter-style glyph selection	**Shaping**, Contextual glyph selection
Horizontal advances only	2D glyph placement
No control over typographic features	Full typographic feature support
Fails on scripts that require contextual shaping	Comprehensive script and language support

Recently in Text & Layout

- Switching to Complex Text
- Word Cache
- Shaper-driven Segmentation
- Improving Font Fallback
- Specification Work & Improvements in CSS3 Writing-modes & Fonts

Switching to Complex

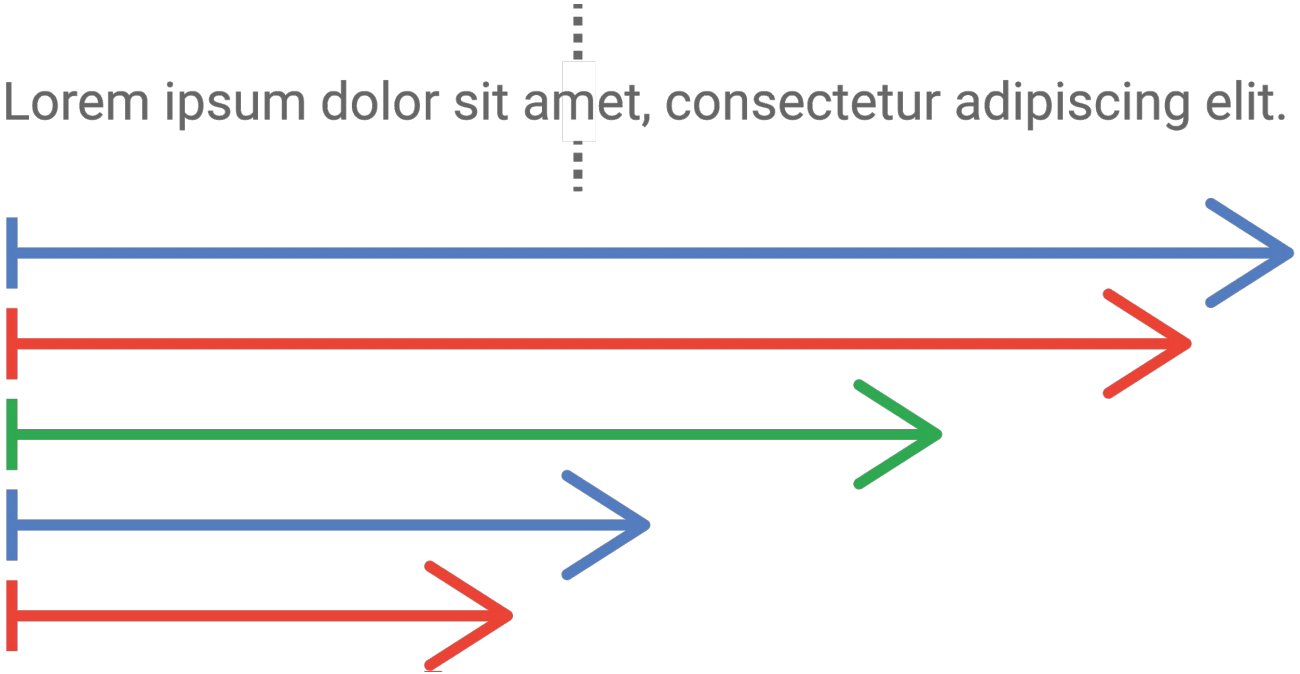
Switching to Complex

- No more rendering difference bugs: simple vs. complex
- Unified code
- Advanced typography always available
- Complex script languages become first class citizen

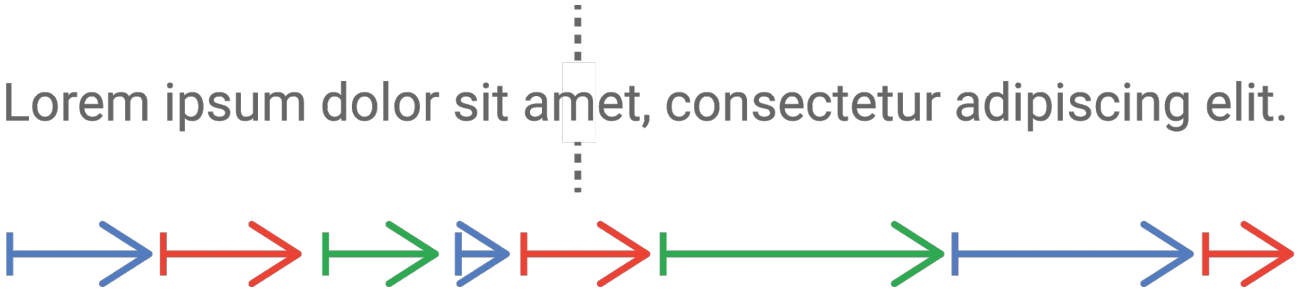
But:

- Full shaping is more costly
- Avoid performance regressions
- Word cache!

Layout of a `TextRun` - No Word Cache



Layout of a `TextRun` - With Word Cache

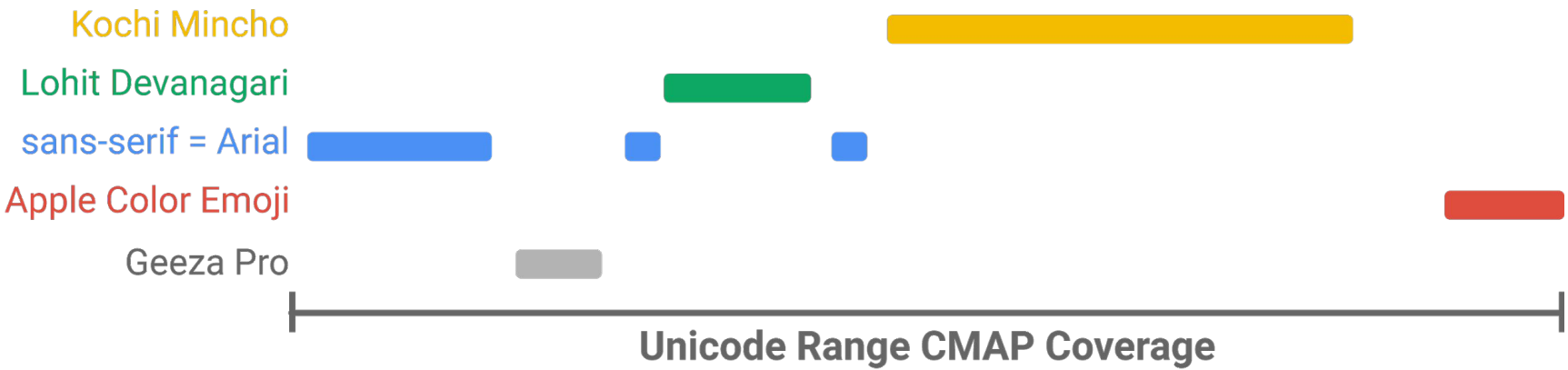


Word Cache aids in `Font.h` operations:
Width, Drawing, Selection

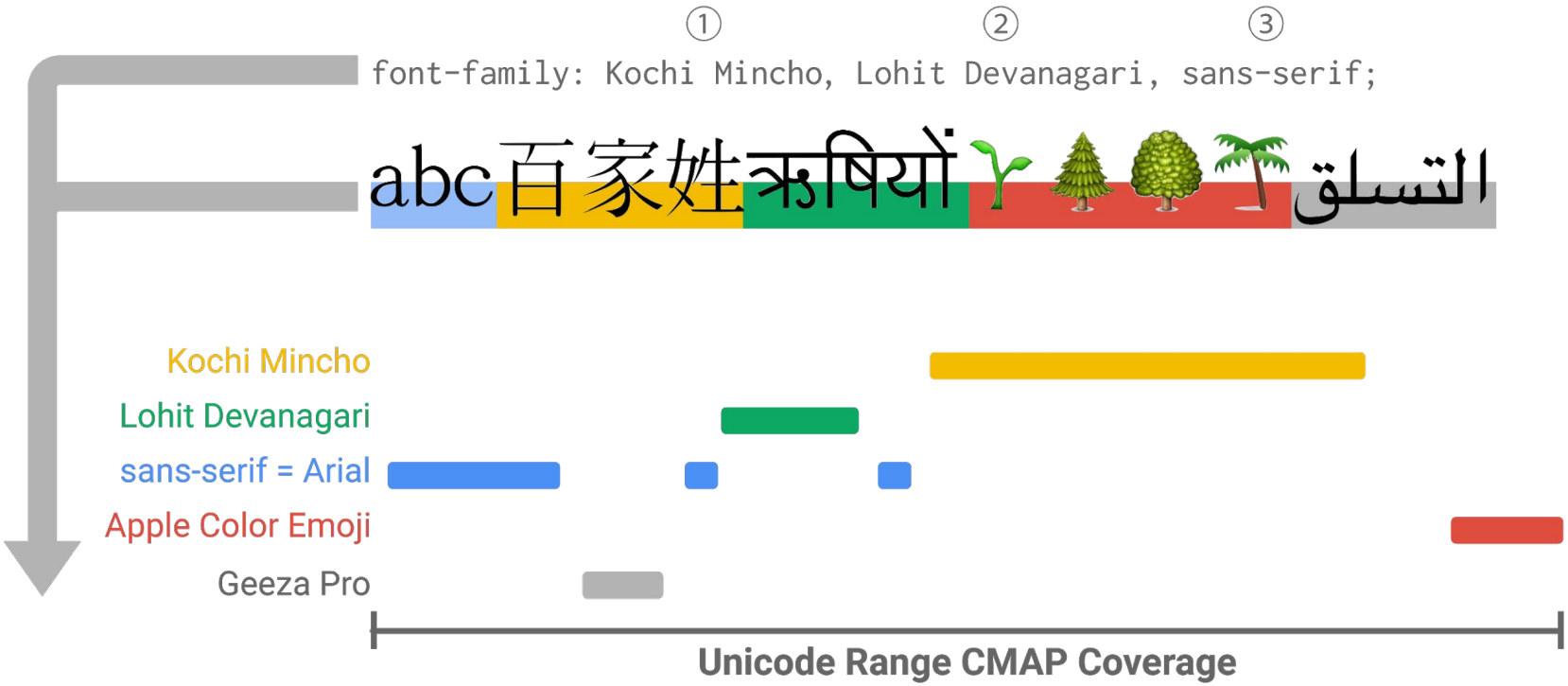
Shaper Driven Segmentation

- Splitting a text run into individual pieces suitable for shaping
- Constant values for the variables of measuring text

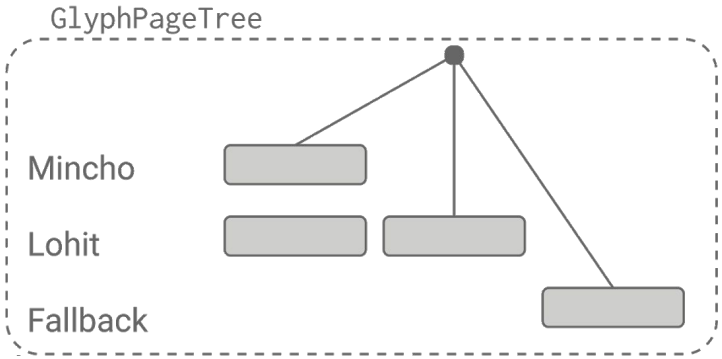
Font Coverage



Font Fallback

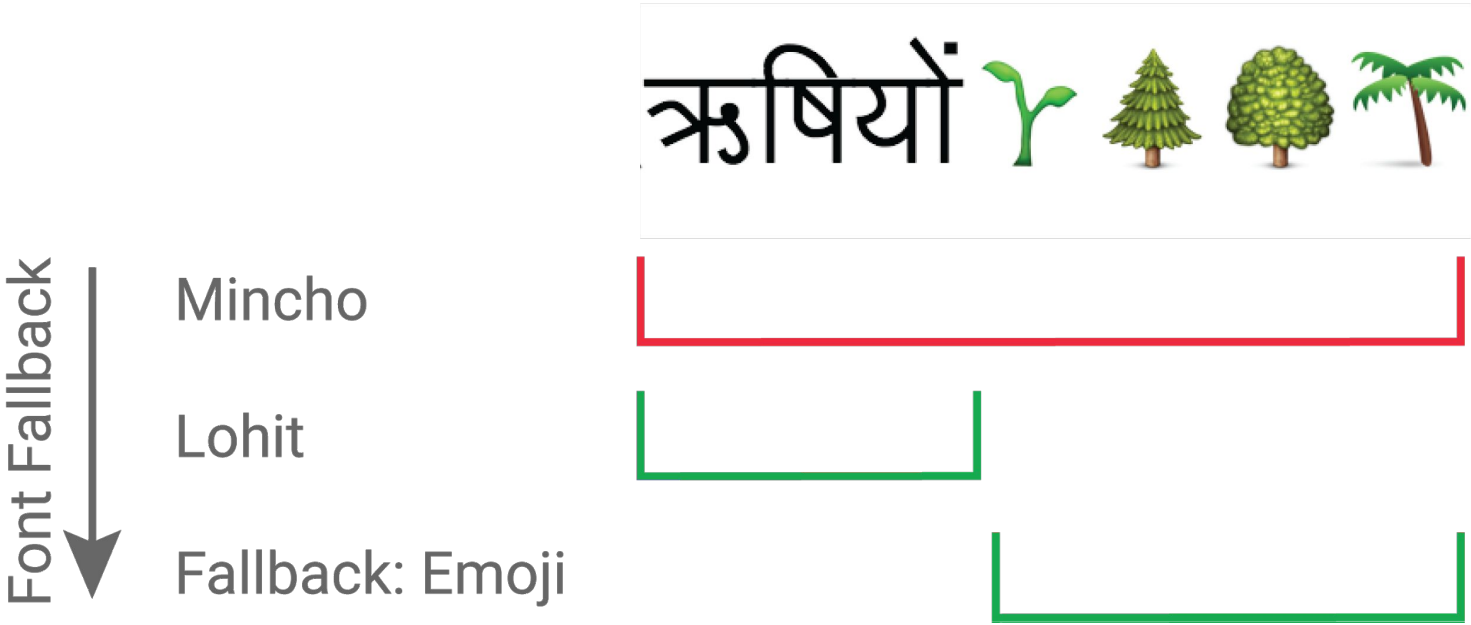


Before: Segmentation through `GlyphPageTree` results



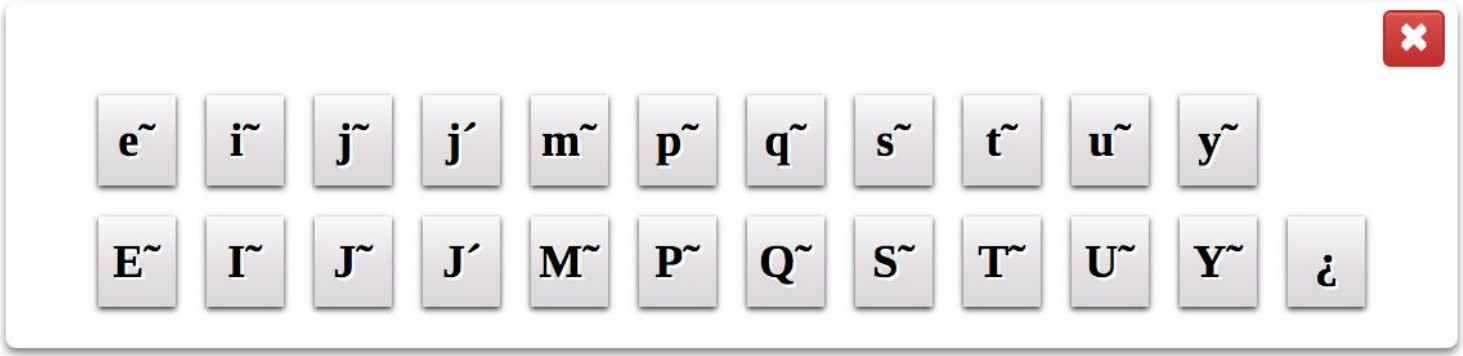
New: Clean Run Segmentation Stage

New: Shaper Driven Segmentation for Font Fallback



Before: Combining Marks

crbug.com/454405, reported by javsmo



Before: Fallback Mixup for Combining Marks

U+0065 LATIN SMALL LETTER E
U+0327 COMBINING CEDILLA
U+0306 COMBINING BREVE

Grapheme Cluster Based Fallback & Font Selection

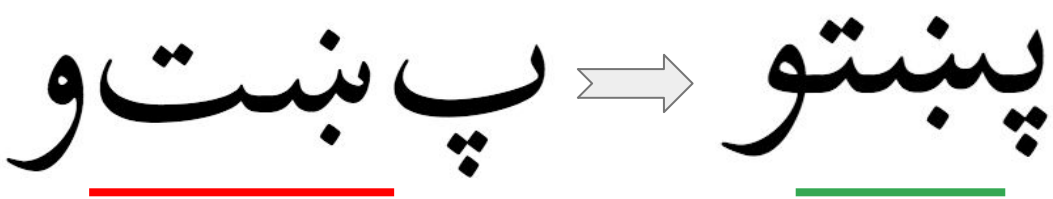
Rendered Fonts

Times — 2 glyphs

Grapheme Cluster Based Fallback & Font Selection

U+0065 LATIN SMALL LETTER E
U+0327 COMBINING CEDILLA
U+0306 COMBINING BREVE

Shaping context across font fallback

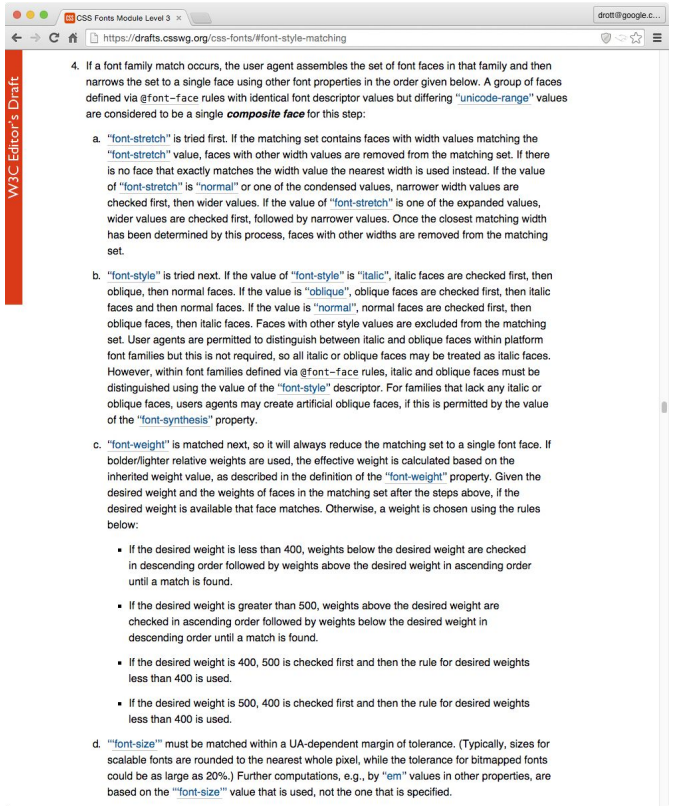



```
unicode-range:
U+069A,U+062A,U+646;
```

Specification Work & Improvements

CSS Font Matching Improvements

- CSS3 Fonts Module Section 5.2 § 4
- Priority order of matching
  - Choose closest **stretch** value, then
  - Choose closest **style** value, then
  - Choose closest **weight**, then
  - Match **size**
- Two main issues:
  - Stretch matching was not implemented, #513670
  - Priority order violated, #513669



CSS Writing Mode Improvements

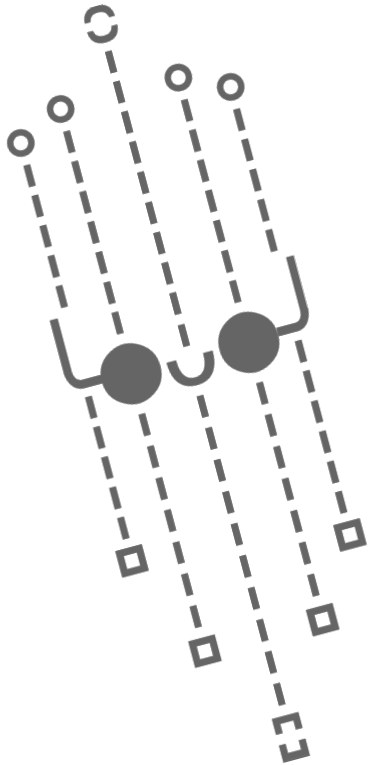
- Discussions in CSS WG on writing-mode attribute values concluded
- Pushed “CSS Writing Modes Level 3” to Candidate Recommendation status
- Intend to unprefix `-webkit-writing-mode:` to `writing-mode:` LGMT’ed
- Unprefixing implemented
- Contributing tests
- Test pass rate up: 85.78% (IE 71.33%, FF 86,24%)
- Spec editorship, contributions to W3C CSS Text, Text Decorations, Writing Modes, Ruby, Unicode UTR#50 Vertical Text Layout

Recently in Text & Layout

- Switching to Complex Text
- Word Cache
- Shaper-driven Segmentation
- Improving Fallback
- Spec Compliance Improvements

Future Plans

- Fully remove Simple Path, GlyphPageTree and friends
- Improve Fallback (System APIs)
- Improve testability
- Improve WordCache caching



Thanks!