



What's New in ECMA-402

Unicode
Technology
Workshop
2025



Shane Carr

Convener, ECMA TC39-TG2

Chair, Unicode ICU4X-TC

Internationalization Engineering @ Google

(this talk is not on behalf of Google)

INTERNATIONALIZATION

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

I18N

#LuisFonsi #Despacito #Imposible

Luis Fonsi - Despacito ft. Daddy Yankee

6,268,533,956 views



33M



4M



SHARE



SAVE

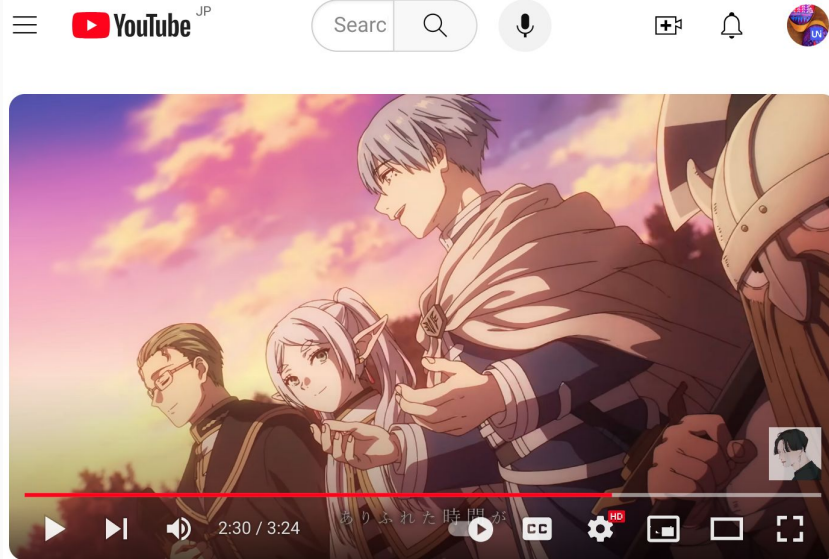


Luis Fonsi

Published on Jan 12, 2017

SUBSCRIBE 25M

i18n in the Wild

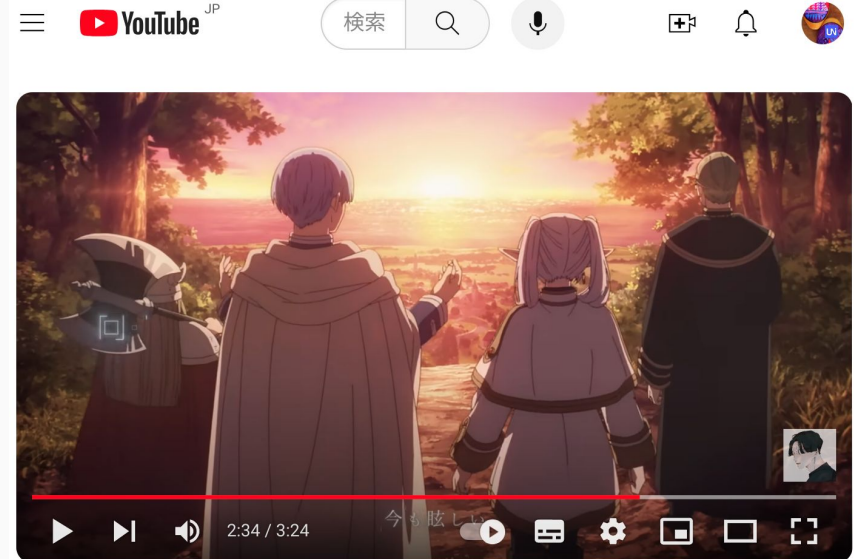


Show chat replay

YOASOBI 「勇者」 Official Music Video / TVアニメ『葬送のフリーレン』オープニングテーマ



3.5M views 2 days ago #3 on Trending for music



チャットのリプレイを表示

YOASOBI 「勇者」 Official Music Video / TVアニメ『葬送のフリーレン』オープニングテーマ

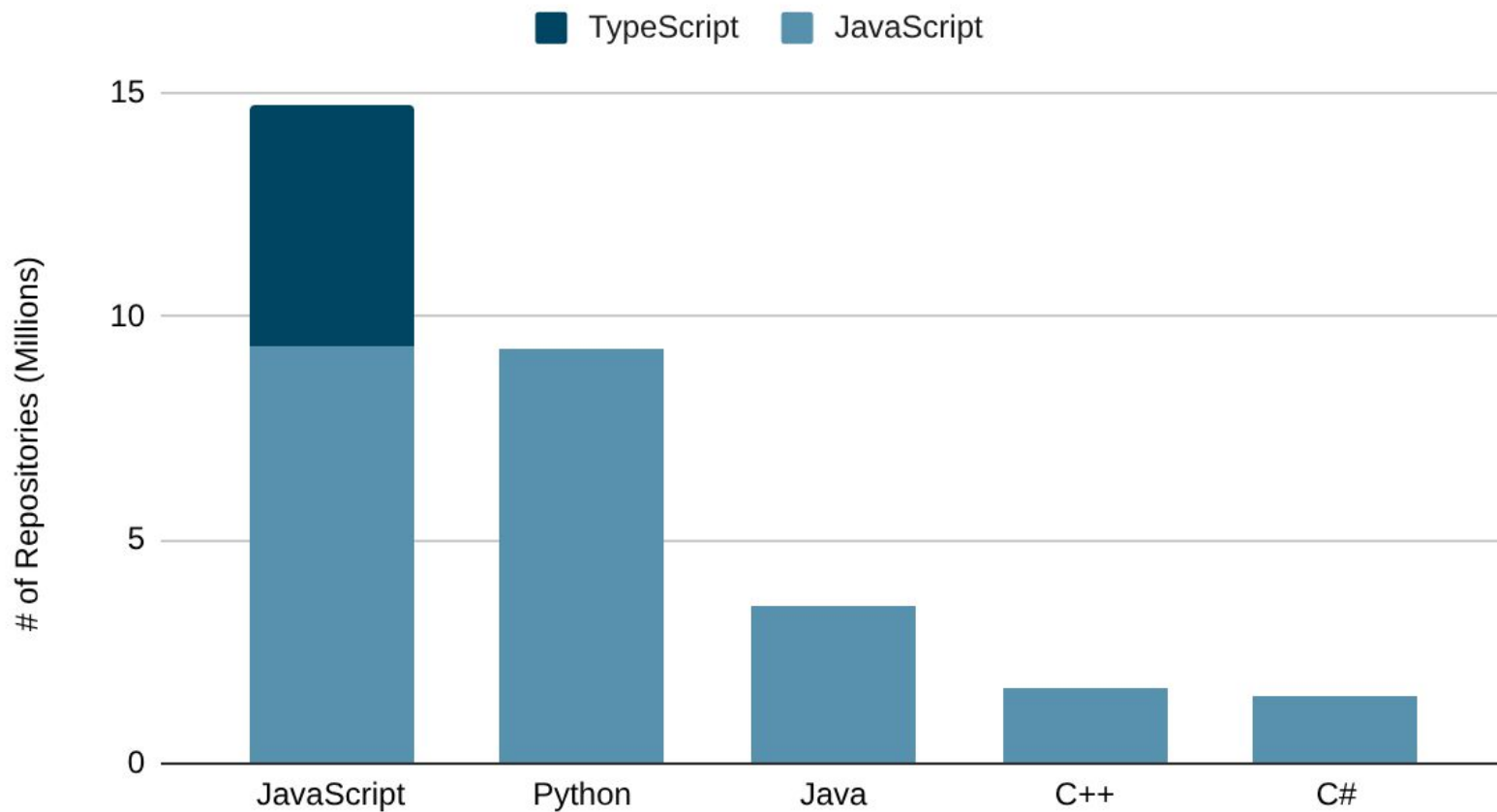


357万 回視聴 2 日前 人気急上昇中の音楽 #3

More Examples of i18n Operations

- BCP-47 Locale Manipulation
 - pl-u-ca-gregory-hc-h12
- Number, Currency, and Unit Formatting
 - ¥10万
- Date and Time Formatting
 - 令和5年10月2日月曜日
- Number and Date Range Formatting
 - 3/4/2023 – 2/10/2023
- Relative Time Formatting
 - 1 day ago
- Temporal and Eras
- Collation
 - ä > z
- Plural Rules
 - zero, one, two, few, many, other
- List Formatting
 - Motocicleta, Autobús o Coche
- Display Names of languages, regions, ...
- Text Segmentation
 - . ユニコード . は . 、 . 符号 . 化 . 文字 . 集合 . や . 文字 . 符号 . 化 . 方式 . など . を . 定め . た .

Most Widely Used Languages on GitHub (2025)



i18n in ECMAScript

TC39-TG2 writes ECMA-402, the standard for built-in JavaScript i18n: Intl.*,
*.prototype.toLocaleString

Promotes best practices and eliminates the need to carry an extra i18n library.

With awesome i18n tools in ECMAScript, we can raise the bar for localization across the industry.



Temporal: Shipping next month in Chrome!

```
let isoDate = Temporal.PlainDate.from("2025-11-11");
let hebrDate = isoDate.withCalendar("hebrew");
let nextDate = hebrDate.add({ months: 1 });

// "2025-12-10[u-ca=hebrew]"
nextDate.toString();

// "20 Kislev 5786"
nextDate.toLocaleString("en", { calendar: nextDate.calendarId })
```

Now Shipping: Intl.DurationFormat 🎉

```
const duration = {  
  hours: 1,  
  minutes: 46,  
  seconds: 40,  
};  
  
// "1 heure, 46 minutes et 40  
// secondes"  
new Intl.DurationFormat("fr-FR", {  
  style: "long"  
}).format(duration);
```

```
const duration = {  
  seconds: 12,  
  milliseconds: 345,  
  microseconds: 600  
};  
  
// "0:00:12.35"  
new Intl.DurationFormat("en", {  
  style: "digital",  
  fractionalDigits: 2  
}).format(duration);
```

Now Shipping: Intl Locale Info

```
let he = new Intl.Locale("he")
let zhHK = new Intl.Locale("zh-HK")
let ar = new Intl.Locale("ar")

// {firstDay: 7, weekend: [5, 6]}
he.getWeekInfo()

// { direction: "rtl" }
let ti = ar.getTextInfo();
```

```
// ["gregory", "chinese"]
zhHK.getCalendars()
```

```
// ["stroke", "big5han",
"gb2312han", "pinyin",
"unihan", "zhuyin",
"emoji", "eor"]
zhHK.getCollations()
```

```
// ["h12"]
zhHK.getHourCycles()
```

```
// ["Asia/Hong_Kong"]
zhHK.getTimeZones()
```

Incubating: Amount

```
let amt = new Amount(2.499, "USD");  
let rounded = amt.roundForCurrency();  
rounded.toLocaleString("en"); // "$2.50"  
rounded.toLocaleString("en-CA"); // "US$2.50"
```

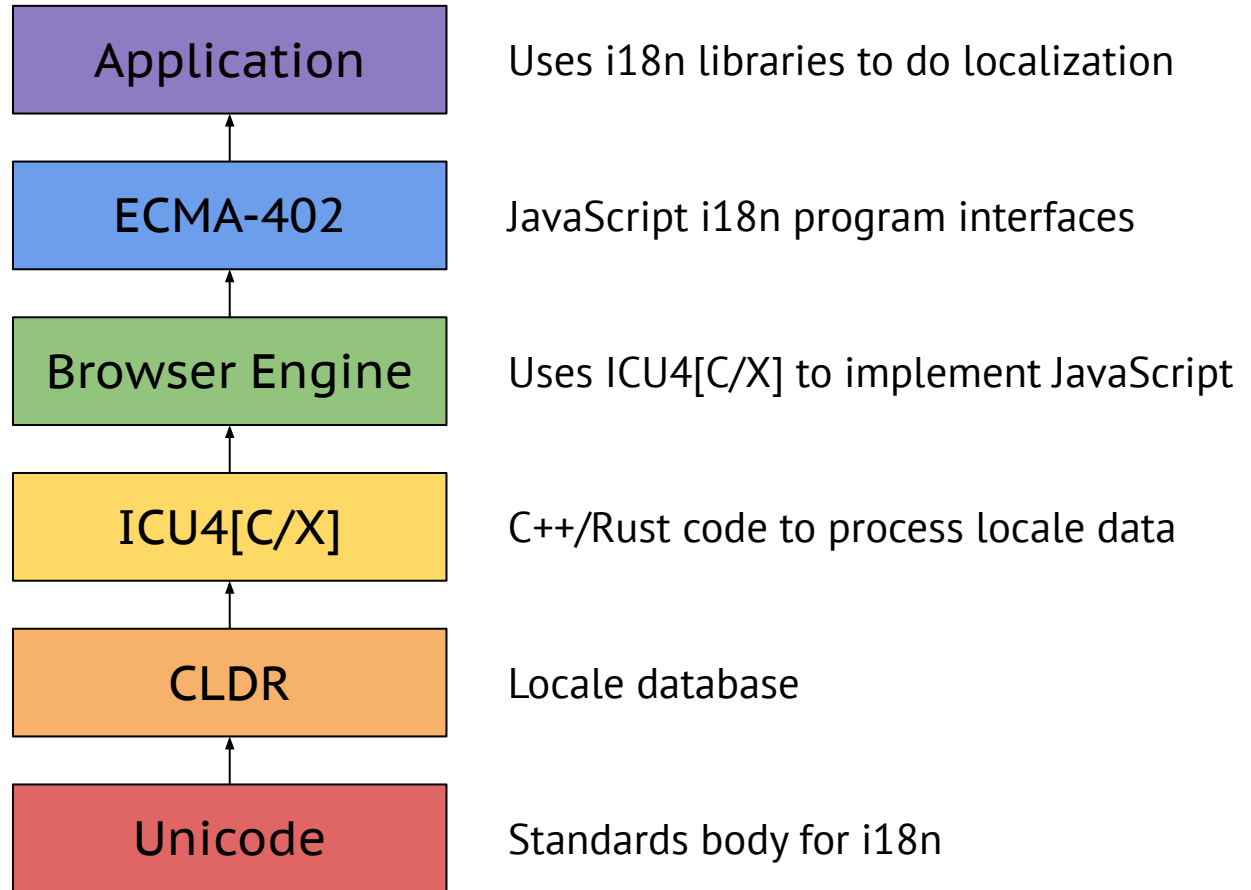
*Do you find this useful? Please
share your input on the proposal!*

Common Misconceptions about i18n

1. Myth: Internationalization is a solved problem
 - Fact: Languages and cultures change; needs of applications evolve
2. Myth: To make my app global-ready I just need to translate the interface
 - Fact: You may need to change layout, text handling, date/time/number formats, collation, ...
3. Myth: I only ship in the US so I only need to support English
 - Fact: 68 million Americans speak a language other than English at home
4. Myth: I18n is not relevant to my industry like Finance or IT
 - Fact: Great i18n can give your product a competitive advantage
5. Myth: I can write my own formatting code and it'll be better/faster
 - Fact: ECMA-402 and ICU4X are small and fast, and they implement an intricate specification
6. Myth: I can consider i18n later when we start shipping in more languages
 - Fact: i18n is an architecture. Build it in early for a better experience for everyone.

[Over 50 more myths!](#)

The i18n Stack



Multiple ways to get your i18n!

Platform ECMA-402

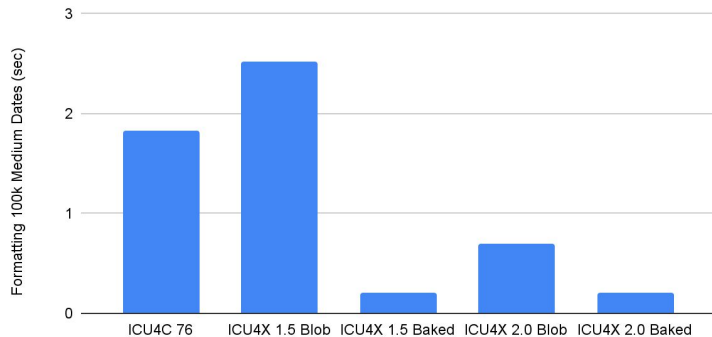
1. Reduced code size and dependencies
2. Consistent i18n across apps on the same device
3. Easy to do the right thing

Ship Your Own ICU[4X]

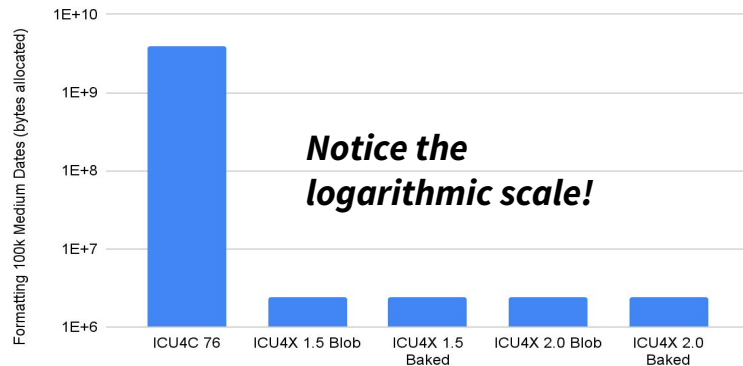
1. More control over app behavior
2. Ability to use locales and features not in the platform
3. Consistent behavior in the same app across devices

ICU4X: i18n that is Portable, Lightweight, and Secure

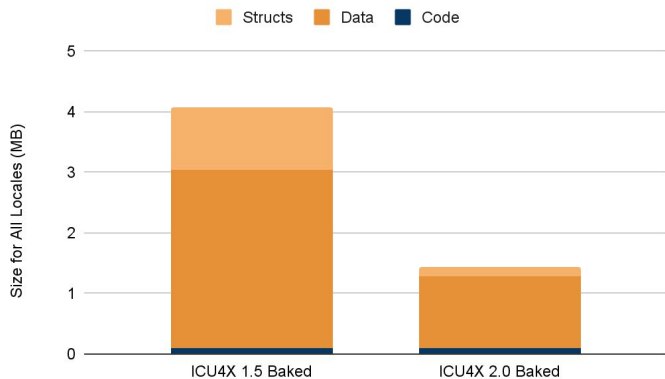
DateTimeFormatter Performance (lower is better)



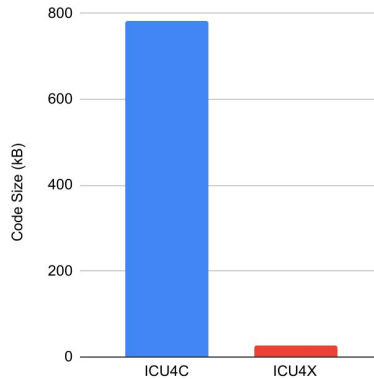
DateTimeFormatter Memory Use (lower is better)



Binary Size: ICU4X 1.5 vs 2.0 (Multi-Component Demo)



Basic Number Format: Code Size



Call to Action

How to contribute:

- Documentation (tech writing)!
- Lots of opportunities for mentorship in spec writing or Rust code
- More i18n and l10n professionals to bring additional perspectives

ICU4X docs: icu4x.unicode.org

Join ECMA-402 monthly
teleconference: contact
ecma402-admin@chromium.org

Let's keep in touch!

shane@unicode.org

