Engineer

Інтернаціоналізація

проєктування програми так, щоб її легко було локалізувати

Що це таке

Інтернаціоналізація — це інженерна робота, завдяки якій один продукт працює багатьма мовами й у багатьох регіонах без змін у коді під кожен із них. W3C визначає її як таке проєктування й розроблення продукту, застосунку чи документа, що дає змогу легко локалізувати його для аудиторій, які різняться культурою, регіоном або мовою. Англійською слово пишуть internationalization або internationalisation і скорочують до i18n, бо між першою і та останньою n стоїть 18 літер; це скорочення вживають і в українських кодових базах. Дехто називає те саме глобалізацією.

W3C зводить розподіл праці до одного речення: дизайнери й розробники інтернаціоналізують продукти, щоб перекладачі та фахівці з локалізації могли легко їх локалізувати. Інтернаціоналізацію роблять один раз, у коді й шаблонах. Локалізацію роблять заново для кожного ринку, у даних, які читає код. Продукт може бути повністю інтернаціоналізований і все одно виходити лише однією мовою.

Обидві спираються на локаль. Специфікація CLDR визначає локаль як ідентифікатор набору вподобань користувача, які зазвичай спільні для великих частин світу: це мова, часто уточнена регіоном, тож `en-GB` і `en-US` мають спільну мову, але по-різному записують дату. Ідентифікатор водночас обирає і переклад, і набір місцевих норм.

W3C поділяє цю роботу на чотири види. Перший усуває перешкоди: продукт обробляє текст як Unicode або свідомо працює зі старим кодуванням там, де без нього не обійтися, не складає речень зі шматків і ніде в коді не залежить від того, які слова бачить користувач. Другий додає підтримку, яка знадобиться лише пізнішій мові: розмітку для тексту, де змішано письмо справа наліво й зліва направо, спосіб позначити, якою мовою написано

фрагмент, і стилі для вертикального тексту. Третій дає коду змогу користуватися місцевими нормами, зазвичай із наявних бібліотек, а не з таблиць, які команда пише сама: форматами дати й часу, календарями, форматами чисел і системами числення, порядком сортування списків, правилами для власних імен і форм звертання. Четвертий виносить усе, що підлягає перекладу, за межі вихідного коду, щоб для кожного читача можна було завантажити потрібну версію.

Застосуйте поняття до рішення

Застосуйте «Що таке інтернаціоналізація?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Програма, що природно читається на кожному ринку, де ви продаєте, — з правильними датами, валютами й сортуванням імен.

Переглянути наші послуги Зв'язатися з нами →

Перевірте припущення за терміном

Використайте пояснення та пов'язані роботи, щоб сформулювати запитання для проєкту. Ми можемо перевірити припущення разом до вибору інструмента. Створила англо-данську систему i18n — 12,027 повідомлень на локаль — англійською та данською

Створила англо-данську систему i18n Зв'язатися з нами →

Чому це важливо

Аргумент W3C стосується часу. Переробити продукт, збудований навколо однієї мови й однієї культури, набагато складніше й довше, ніж від початку проєктувати його для світової аудиторії, і W3C порівнює цю роботу з проблемою 2000 року, коли довелося переробляти двоцифрові поля року, що виходили з того, ніби кожен рік починається з 19. Запізніла інтернаціоналізація зазвичай перетворюється на незграбну й дорогу переробку. Рання зводиться до кількох звичок, які майже нічого не коштують того дня, коли їх запроваджують.

Ціна відсутньої звички зростає з кожною новою мовою. Речення, яке код складає з трьох шматків, — це одна вада в англомовному продукті й п’ять вад у продукті з п’ятьма локалями, і щоразу виправленням буде не кращий переклад, а зміна в коді, яку переглядають і випускають, як будь-яку іншу. Те саме стосується дати, відформатованої вручну, списку, відсортованого порівнянням байтів, і тесту, що шукає кнопку за англійським написом.

Звички окупаються навіть тоді, коли нічого так і не перекладають, і W3C наголошує на цьому прямо. Продукт, який продають лише українською, однаково отримує імена з літерами, яких немає в українській абетці, адреси,

записані за звичаєм іншої країни, і дати із систем, що ставлять місяць перед днем. Текст, збережений як Unicode, і дані, у яких дата лишається датою аж до показу, впораються з усім цим без жодних особливих випадків.

Як це виглядає на практиці

Найшвидша перевірка того, чи продукт інтернаціоналізовано, — речення з числом. Неінтернаціоналізована версія складає його просто в коді:

```
const label = count + (count === 1 ? " day" : " days");
```

Англійською це правильно, а перекласти так українською неможливо. Для цілих чисел українська має три форми, а для дробових четверту: «день» після 1, 21 і 101, «дні» після 2, 3, 4 і 22, «днів» після 0, від 5 до 20 і 25, «дня» після дробового числа. Хоч яке одне слово перекладач поставить у другу гілку, для більшості чисел воно буде хибним, бо 21 вимагає тієї самої форми, що й 1, а 2 іншої, ніж 5.

Після інтернаціоналізації речення стає ключем, а таблиця кожної мови містить усі форми, які цій мові потрібні. У Hugo таблиці перекладів лежать у каталозі `i18n`, по одному файлу на мову, названому за мовним тегом:

```
# i18n/en.toml
[days]
one = '{{ . }} day'
other = '{{ . }} days'
```

```
# i18n/uk.toml
[days]
one = '{{ . }} день'
few = '{{ . }} дні'
many = '{{ . }} днів'
other = '{{ . }} дня'
```

Шаблон запитує ключ і передає число. У ньому немає жодного слова ні англійською, ні українською і жодного правила про множину:

```
{{ T "days" 21 }}
```

Англійські сторінки показують `21 days`, українські `21 день`; з 2 це `2 days` і `2 дні`, з 5 це `5 days` і `5 днів`. Назви `one`, `few`, `many` і `other` вигадав не Hugo. Це категорії множини, визначені в CLDR, зібранні даних про локалі, яке веде Консорціум Unicode, а документація Hugo відсилає до таблиці правил множини CLDR, щоб дізнатися, які категорії має кожна мова. Нова мова з'являється як новий файл, без жодної зміни в шаблонах.

Коли продукт росте, таблиці стають найбільшим масивом тексту в ньому й потребують тієї самої дисципліни, що й код. Один із проєктів цього сайту, англо-данська система i18n, містить 12 027 повідомлень на локаль у 458 файлах простору імен. Кожен файл обмежено 400 рядками, лінтер звіряє ключі між локалями, щоб повідомлення, наявне в одній, не було відсутнім в іншій, а для данської зафіксовано словник, аж до неформального звертання «du», і його дотримання інструментарій гарантує на кожному коміті. Головною опорою сторінка називає саме ліміт: без обмеження на файл замість 458 файлів було б дванадцять.

Де виникають проблеми

Речення склеюють зі шматків. Повідомлення, складене як `"Delete " + n + " files?"`, доходить до перекладача двома уривками без речення навколо і з порядком слів, потрібним лише англійській. Інша мова може вимагати числа наприкінці або форми дієслова, що залежить від іменника. W3C прямо називає обережність зі склеюванням рядків частиною цієї роботи. Робіть кожне речення одним повідомленням з іменованими заповнювачами, які перекладач може переставляти; документація Hugo показує, як `{{ T "age" (dict "name" "Will" "count" 1) }}` підставляє і ім'я, і число в одне перекладене речення.

Код читає слова з екрана. Тест, що шукає кнопку за написом «Зберегти», чи гілка, яка порівнює статус із текстом «Оплачено», працює до першого перекладу, а потім ламається без жодного повідомлення про помилку. W3C серед перших пунктів інтернаціоналізації називає відмову від будь-якої залежності коду від рядків інтерфейсу. Порівнюйте ключі, ідентифікатори й значення переліків, а тестам давайте стабільний атрибут, за яким шукати елемент.

Одне правило множини для всіх мов. `count === 1` — це англійське правило, записане кодом, і для більшості мов, які може додати продукт, воно хибне. Специфікація CLDR визначає шість категорій множини: zero, one, two, few, many і other. Обов'язкова з них лише other, а для кожної мови специфікація задає власні правила вибору: в англійській категорій дві, в українській чотири. Передавайте число в повідомлення, а таблиця перекладу хай містить кожну категорію, якою користується мова.

Відсутній переклад зникає без звуку. Функція `T` у Hugo, не знайшовши ключа в поточній мові, бере значення з мови за замовчуванням, а якщо немає й там, повертає порожній рядок. Тож сторінка може вийти з прогалиною або з англійською фразою посеред української. Збирайте сайт із `--printI18nWarnings`, щоб отримати список усіх відсутніх перекладів і резервних значень, увімкніть `enableMissingTranslationPlaceholders` у тестовому середовищі, щоб прогалини було видно на сторінці, і звіряйте ключі всіх локалей у безперервній інтеграції.

Дати й числа форматують вручну. Шаблон, що пише `MM/DD/YYYY`, або число з комою після кожних трьох цифр жорстко прописують у коді одну країну. Той самий день записують як `10/27/26` в американській англійській, `27/10/2026` у британській, `27.10.2026` у данській і `2026-10-27` у шведській, а число, яке українською пишуть 12 027,5, англійською має вигляд `12,027.5`, а данською `12.027,5`. Специфікація CLDR радить зберігати й передавати дані в мовно нейтральному вигляді й форматувати їх уже для читача. Тримайте дати й суми як прості значення до останнього кроку і форматуйте їх там функціями платформи, що враховують локаль. У Hugo `time.Format ":date_full"` показує ту саму дату як `Wednesday, November 3, 2021` на англійській сторінці й `Mittwoch, 3. November 2021` на німецькій.

Текст зберігають не як Unicode. Слово «пам'ять», записане в UTF-8 і прочитане як Windows-1251, перетворюється на «РїР°Рј'СЏС‚СЊ», а продукт, що зберігає текст у старому однобайтовому кодуванні для кирилиці, не вмістить данського прізвища з літерою ø. Підтримка Unicode стоїть першою в переліку W3C. Використовуйте одне кодування Unicode, зазвичай UTF-8, на всьому шляху від бази даних до браузера й оголошуйте його на кожній межі, де байти стають текстом.

Що почитати далі

- Localization vs. Internationalization — робочі визначення обох понять від W3C, чотири види роботи, з яких складається інтернаціоналізація, і чому її місце на самому початку.
- Hugo: Multilingual mode — таблиці перекладів, локалізовані дати й валюти і те, як Hugo повідомляє про відсутній переклад.
- Hugo: lang.Translate — порядок пошуку перекладу між мовами, категорії множини в таблицях перекладів і повідомлення з іменованими заповнювачами.
- UTS #35, Part 1: Core — специфікація CLDR: ідентифікатори локалей, успадкування між ними і порада тримати дані мовно нейтральними аж до показу.
- UTS #35, Part 3: Numbers — символи й шаблони чисел для кожної локалі, а також категорії множини та правила вибору між ними.



Словник

Обирайте за фактичними умовами

Опишіть робочий процес, наявні інтерфейси та користувачів, яким потрібна зміна. Ми можемо визначити перевірюваний результат і спосіб передачі, який дасть вашій команді змогу підтримувати його.

Дивитися, скільки це коштує Зв'язатися з нами →

Зробіть термінологію практичною

Запитайте, як «Що таке інтернаціоналізація?» стосується вашої організації. Конкретного прикладу з поточної системи достатньо для початку розмови.

Переглянути наші послуги

welcome@engineer.company +45 50 10 61 84 Копенгаген, Данія

Візьміть цю сторінку із собою

Візитівка

Дбайливо зроблено з любов'ю — D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Авторське право © 2025 – дотепер · Engineer ApS · Усі права захищено.

Надруковано з https://engineer.company/uk/glossary/internationalization/