Engineer

Internationalisierung

Software so gestalten, dass sie leicht lokalisierbar ist

Was es ist

Internationalisierung ist die technische Arbeit, durch die ein Produkt in vielen Sprachen und Regionen funktioniert, ohne dass sein Code für jede davon geändert werden muss. Das W3C definiert sie als das Entwerfen und Entwickeln eines Produkts, einer Anwendung oder eines Dokuments mit dem Ziel, eine spätere Lokalisierung unabhängig von Kultur, Region und Sprache des Publikums zu erleichtern. Das englische Wort lautet internationalization oder internationalisation. Die Kurzform i18n steht für die 18 Buchstaben zwischen dem ersten i und dem letzten n. Manche nennen denselben Gedanken Globalisierung.

Das W3C fasst die Arbeitsteilung in einem Satz zusammen: Wer Produkte entwirft und entwickelt, internationalisiert sie, „damit Übersetzer und Lokalisierungsfachleute sie leicht lokalisieren können“. Internationalisierung erfolgt einmal, im Code und in den Vorlagen. Lokalisierung erfolgt für jeden Markt erneut, in Daten, die der Code einliest. Ein Produkt kann vollständig internationalisiert sein und trotzdem nur in einer Sprache ausgeliefert werden.

Die Einheit, mit der beide arbeiten, ist das Gebietsschema, auch Locale genannt. Die CLDR-Spezifikation definiert eine Locale als Kennung für Präferenzen, die Nutzer in großen Teilen der Welt gewöhnlich gemeinsam haben: eine Sprache, häufig durch eine Region eingegrenzt. So haben `en-GB` und `en-US` dieselbe Sprache, unterscheiden sich aber in der Schreibweise eines Datums. Die Kennung wählt gleichzeitig eine Übersetzung und einen Satz von Konventionen aus.

Das W3C nennt vier Arten von Arbeit unter diesem Begriff. Die erste beseitigt Hindernisse: Das Produkt verarbeitet Text als Unicode oder behandelt eine ältere Zeichenkodierung dort bewusst, wo sie erforderlich ist; es setzt Sätze nicht aus Fragmenten zusammen, und sein Code hängt nie vom angezeigten Wortlaut ab.

Die zweite ergänzt Unterstützung, die erst eine spätere Sprache benötigt, etwa Auszeichnung für Text, der Schriften mit Leserichtung von rechts nach links und von links nach rechts mischt, eine Möglichkeit, die Sprache einer Passage anzugeben, sowie Stilregeln für vertikalen Text. Die dritte ermöglicht dem Code, lokale Konventionen zu verwenden, üblicherweise aus vorhandenen Bibliotheken statt aus selbst geschriebenen Tabellen: Datums- und Zeitformate, Kalender, Zahlenformate und Ziffernsysteme, die Sortierreihenfolge von Listen sowie Konventionen für Personennamen und die Anrede. Die vierte hält alles Übersetzbare aus dem Quellcode heraus, damit für jeden Leser die richtige Fassung geladen werden kann.

Das Konzept für eine Entscheidung nutzen

Nutzen Sie „Was ist Internationalisierung?“ für eine konkrete Entscheidung. Wir können helfen, das Konzept mit dem System in Verbindung zu bringen, das Sie entwickeln oder einkaufen. Software, die in jedem Ihrer Absatzmärkte natürlich klingt — einschließlich Datumsangaben, Währungen und der Sortierung von Namen.

Unsere Leistungen entdecken Kontakt aufnehmen →

Die Annahmen hinter dem Begriff prüfen

Nutzen Sie die Erklärung und die verlinkten Arbeiten, um Fragen für Ihr Projekt zu formulieren. Wir können die Annahmen gemeinsam mit Ihnen prüfen, bevor ein Werkzeug ausgewählt wird. Englisch-dänisches i18n-System aufgebaut — 12,027 Nachrichten je Sprachversion, auf Englisch und Dänisch

Englisch-dänisches i18n-System aufgebaut Kontakt aufnehmen →

Warum es wichtig ist

Das Argument des W3C betrifft den Zeitpunkt. Ein um eine Sprache und Kultur herum gebautes Produkt für den Weltmarkt umzuarbeiten kostet wesentlich mehr Aufwand und Zeit, als es von Anfang an für die ganze Welt zu entwerfen. Das W3C vergleicht diese Arbeit mit dem Aufwand für die Y2K-Umstellung zweistelliger Jahresfelder, die voraussetzten, dass jedes Jahr mit 19 beginnt. Späte Internationalisierung bedeutet meist umständliche und teure technische Überarbeitung. Früh begonnen besteht sie aus wenigen Gewohnheiten, die sich bei ihrer Einführung mit wenig Aufwand umsetzen lassen.

Fehlt eine dieser Gewohnheiten, wachsen die Kosten mit jeder weiteren Sprache. Ein im Code aus drei Fragmenten zusammengesetzter Satz bedeutet einen Defekt in einem englischen Produkt und fünf Defekte in einem Produkt mit fünf Locales. In jedem Fall erfordert die Behebung eine Codeänderung statt einer besseren Übersetzung. Diese Änderung muss wie jede andere geprüft und veröffentlicht werden. Dasselbe gilt für ein von Hand formatiertes Datum, eine durch Bytevergleiche sortierte Liste oder eine Beschriftung, die ein Test anhand ihres englischen Texts findet.

Diese Gewohnheiten zahlen sich auch aus, wenn nie etwas übersetzt wird; das W3C sagt das ausdrücklich. Ein ausschließlich auf Englisch verkauftes Produkt erhält trotzdem Namen mit Buchstaben, die das Englische nicht verwendet, nach den Regeln anderer Länder aufgebaute Adressen und Datumsangaben aus Systemen, die den Tag vor den Monat schreiben. Mit als Unicode gespeichertem Text und Daten, die ein Datum bis zur Anzeige als Datumswert bewahren, lässt sich all das ohne Sonderbehandlung verarbeiten.

Wie es in der Praxis aussieht

Der schnellste Test auf Internationalisierung ist ein Satz mit einer Zahl. Die nicht internationalisierte Fassung baut den Satz im Code zusammen:

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

Das ist auf Englisch korrekt und lässt sich nicht übersetzen. Ukrainisch benötigt drei Formen für ganze Zahlen und eine vierte für Brüche: день nach 1, 21 und 101, дні nach 2, 3, 4 und 22, днів nach 0, 5 bis 20 und 25 sowie дня nach einem Bruch. Welches einzelne Wort ein Übersetzer auch in den zweiten Zweig einsetzt, für die meisten Zahlen ist es falsch, denn 21 verlangt dieselbe Form wie 1 und 2 eine andere Form als 5.

Internationalisiert wird der Satz zu einem Schlüssel, und die Tabelle jeder Sprache liefert sämtliche benötigten Formen. In Hugo liegen Übersetzungstabellen im Verzeichnis `i18n` : eine Datei je Sprache, benannt nach ihrem Sprachkennzeichen:

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

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

Die Vorlage fragt den Schlüssel ab und übergibt die Zahl. Sie enthält kein Wort einer der beiden Sprachen und keine Pluralregel:

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

Die englischen Seiten geben `21 days` aus, die ukrainischen `21 день`; bei 2 sind es `2 days` und `2 дні`, bei 5 `5 days` und `5 днів`. Die Namen `one`, `few`, `many` und `other` hat Hugo nicht erfunden. Es sind die Pluralkategorien von CLDR, der Sammlung von Locale-Daten des Unicode Consortiums. Hugos Dokumentation verweist für die Kategorien jeder Sprache auf die Übersicht der CLDR-Pluralregeln. Für eine neue Sprache kommt eine neue Datei hinzu; die Vorlage muss dafür nicht geändert werden.

In großem Maßstab werden die Tabellen zur größten Textmenge des Produkts und benötigen dieselbe Disziplin wie Code. Ein Projekt dieser Website, ein englisch-dänisches Internationalisierungssystem, enthält 12.027 Meldungen je Locale in 458 Namensraumdateien. Jede Datei ist auf 400 Zeilen begrenzt. Ein Linter vergleicht die Schlüssel beider Locales, damit keine Meldung nur in einer vorkommt. Die dänische Seite folgt einem festgelegten Wortschatz bis hin zum informellen „du“, den die Werkzeuge bei jedem Commit durchsetzen. Die Projektseite schreibt die Wartbarkeit dieser Begrenzung zu: Ohne Grenze je Datei, heißt es dort, wären aus den 458 Dateien zwölf geworden.

Wo es schiefgeht

Sätze werden zusammengeklebt. Eine als `"Delete " + n + " files?"` gebaute Meldung erreicht den Übersetzer als zwei Fragmente ohne den umgebenden Satz und in einer Reihenfolge, die nur Englisch benötigt. Eine andere Sprache verlangt die Zahl vielleicht am Ende oder eine vom Substantiv abhängige Verbform. Das W3C nennt einen sorgfältigen Umgang mit der Verkettung von Zeichenketten als Teil der Arbeit. Machen Sie jeden Satz zu einer Meldung mit benannten Platzhaltern, die ein Übersetzer verschieben kann. Hugos Dokumentation zeigt, wie `{{ T "age" (dict "name" "Will" "count" 1) }}` einen Namen und eine Anzahl in einen einzigen übersetzten Satz einsetzt.

Der Code liest die Wörter auf dem Bildschirm. Ein Test, der einen Button anhand der Beschriftung „Save“ findet, oder ein Zweig, der einen Status mit dem Text „Paid“ vergleicht, funktioniert bis zur ersten Übersetzung und versagt dann ohne Fehlermeldung. Das W3C zählt die Vermeidung jeglicher Abhängigkeit des

Codes von Oberflächentexten zu den ersten Aufgaben der Internationalisierung. Vergleichen Sie Schlüssel, Kennungen und Aufzählungswerte und geben Sie Tests ein stabiles Attribut, anhand dessen sie ein Element finden können.

Eine Pluralregel für alle Sprachen. `count === 1` ist die englische Regel als Code und für die meisten Sprachen falsch, die ein Produkt später unterstützen könnte. Die CLDR-Spezifikation definiert sechs Pluralkategorien, zero, one, two, few, many und other, von denen nur other vorgeschrieben ist. Jede Sprache hat eigene Regeln für die Auswahl: Englisch verwendet zwei, Ukrainisch vier. Übergeben Sie die Zahl an die Meldung und lassen Sie die Übersetzungstabelle jede Kategorie enthalten, die die Sprache verwendet.

Eine fehlende Übersetzung bleibt unbemerkt. Hugos `T` greift auf die Standardsprache zurück, wenn ein Schlüssel in der aktuellen Sprache fehlt, und liefert eine leere Zeichenkette, wenn auch die Standardsprache ihn nicht hat. So kann eine Seite mit einer Lücke oder einer englischen Wendung mitten im Dänischen ausgeliefert werden. Lassen Sie beim Build mit `--printI18nWarnings` jede fehlende Übersetzung und jeden Rückgriff auflisten. Aktivieren Sie in einer Vorschauumgebung `enableMissingTranslationPlaceholders`, damit Lücken auf der Seite sichtbar werden, und vergleichen Sie die Schlüssel aller Locales in der kontinuierlichen Integration.

Datumsangaben und Zahlen werden von Hand formatiert. Eine Vorlage, die `MM/DD/YYYY` schreibt oder alle drei Ziffern ein Komma einfügt, kodiert ein Land fest ein. Derselbe Tag lautet im amerikanischen Englisch `10/27/26`, im britischen Englisch `27/10/2026`, auf Dänisch `27.10.2026` und auf Schwedisch `2026-10-27`. Die Zahl 12,027.5 wird auf Dänisch als `12.027,5` geschrieben, auf Norwegisch, Schwedisch und Ukrainisch mit einem Leerzeichen als Tausendertrennzeichen und einem Komma vor den Dezimalstellen. Die CLDR-Spezifikation empfiehlt, Daten sprachneutral zu speichern und auszutauschen und sie für den Leser zu formatieren. Behalten Sie Datumsangaben und Beträge bis zum letzten Schritt als unformatierte Werte bei und formatieren Sie sie erst dort mit den Locale-fähigen Funktionen der Plattform. In Hugo gibt `time.Format ":date_full"` ein Datum auf einer englischen Seite als `Wednesday, November 3, 2021` und auf einer deutschen als `Mittwoch, 3. November 2021` aus.

Text wird nicht als Unicode gespeichert. Ein Produkt, das „blåbærsyltetøj“ in UTF-8 schreibt und als Windows-1252 wieder einliest, zeigt „blÃ¥bÃ¦rsyltetÃ¸j“. Speichert es Text in einer älteren, auf eine Sprache beschränkten Kodierung, kann es einen ukrainischen Namen überhaupt nicht aufnehmen. Unicode-

Unterstützung steht an erster Stelle der W3C-Liste. Verwenden Sie von der Datenbank bis zum Browser eine einzige Unicode-Kodierung, normalerweise UTF-8, und deklarieren Sie sie an jeder Grenze, an der Bytes zu Text werden.

Weiterführende Lektüre

- Lokalisierung und Internationalisierung — Die Arbeitsdefinitionen des W3C für beide Begriffe, die vier Arten von Internationalisierungsarbeit und die Gründe, damit am Anfang zu beginnen.
- Hugo: Mehrsprachiger Modus — Übersetzungstabellen, lokalisierte Datumsangaben und Währungen sowie Hugos Meldungen über fehlende Übersetzungen.
- Hugo: lang.Translate — Die Suchreihenfolge über Sprachen hinweg, Pluralkategorien in Übersetzungstabellen und Meldungen mit benannten Platzhaltern.
- UTS #35, Teil 1: Grundlagen — Die CLDR-Spezifikation: Locale-Kennungen, Vererbung zwischen Locales und die Empfehlung, Daten bis zur Anzeige sprachneutral zu halten.
- UTS #35, Teil 3: Zahlen — Zahlensymbole und Muster je Locale sowie die Pluralkategorien mit ihren Auswahlregeln.

Glossar

Für Ihre tatsächlichen Rahmenbedingungen auswählen

Beschreiben Sie den Arbeitsablauf, die vorhandenen Schnittstellen und die Nutzer, die die Veränderung benötigen. Wir können eine prüfbare Lieferung definieren und die Übergabe so gestalten, dass Ihr Team die gelieferten Bestandteile warten kann.

Preise ansehen Kontakt aufnehmen →

Die Begriffe praktisch nutzbar machen

Fragen Sie, wie „Was ist Internationalisierung?“ auf Ihre Organisation zutrifft. Ein konkretes Beispiel aus Ihrem bestehenden System reicht aus, um das Gespräch zu beginnen.

Unsere Leistungen entdecken

welcome@engineer.company +45 50 10 61 84 Kopenhagen, Dänemark

Nehmen Sie diese Seite mit

Kontaktkarte

Mit Sorgfalt und Liebe gestaltet von D.

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

Copyright © 2025 – heute · Engineer ApS · Alle Rechte vorbehalten.

Gedruckt von https://engineer.company/de/glossary/internationalization/