

Fachvortrag

I18n

Software Engineering 2

SS19

05.07.2019

Ethnic Blue

Gliederung

1. Einleitung
2. G11n, I18n, L10n und T9n
3. ISO & Standardisierung
4. Java
5. HTML & XML
6. WordPress
7. Beispiele
 - 7.1. JavaScript
 - 7.2. Nivea / McDonald's
8. Fazit (Vorteile/Nachteile)
9. Quellen

Einleitung

Unternehmen wollen:

- ihre Website, Anwendung oder andere Software für den globalen Markt erstellen
- Verbraucher aus bestimmten Ländern ansprechen
- Produkt entsprechend dieser Kultur und Sprache anpassen

Begriffe:

- Globalization (G11n)
- Internationalization (I18n)
- Localization (L10n)
- Translation (T9n)

G11n, I18n, L10n und T9n

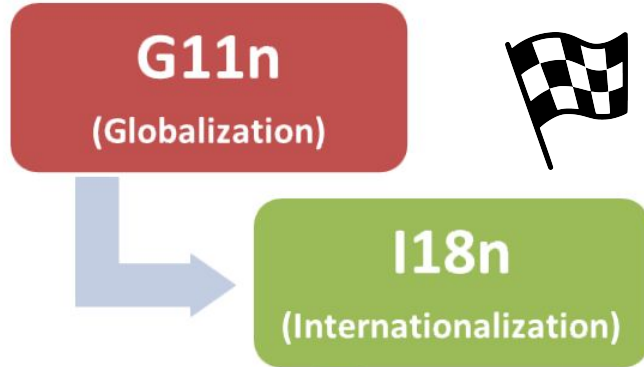
G11n

(Globalization)

Globalization (G11n)

- 11 Buchstaben zwischen G und n
- umfasst ...
 - Internationalization
 - Localization
 - Translation

G11n, I18n, L10n und T9n

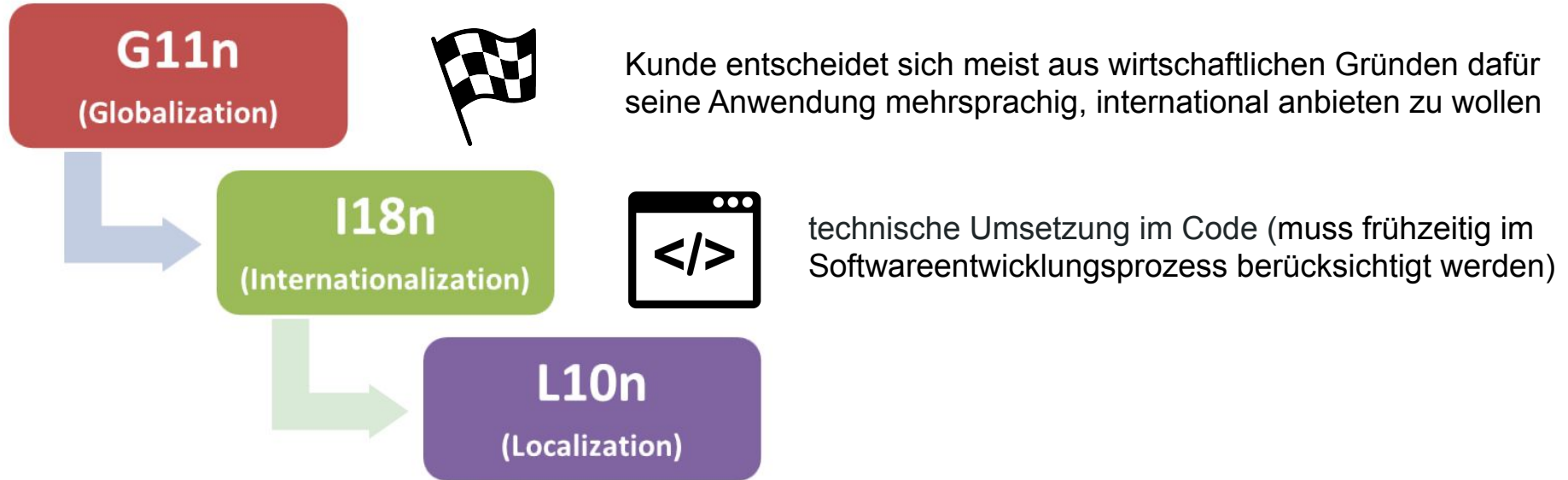


Kunde entscheidet sich meist aus wirtschaftlichen Gründen dafür seine Anwendung mehrsprachig, international anbieten zu wollen

Internationalization (I18n)

- 18 Buchstaben zwischen I und n
- Prozess zur Entwicklung des Quellcodes sodass es leicht an andere Sprachen angepasst werden kann (ohne den Quellcode zu verändern)
 - Felder wie z.B. Text sollten nicht fest codiert und flexibel sein
- Zum Beispiel:
 - Es gibt ...
 - weltweit unterschiedliche Währungssymbole, Format für das Datum, ...

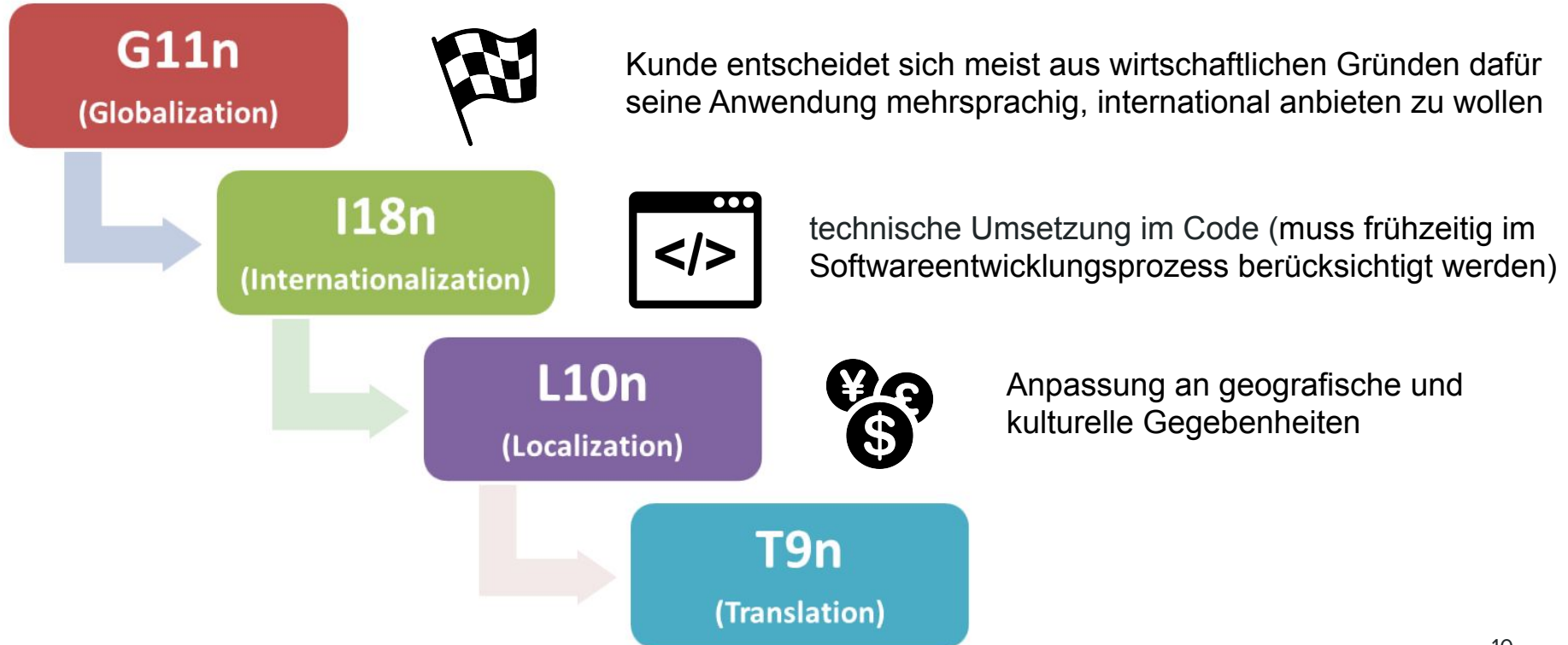
G11n, I18n, L10n und T9n



Localization (L10n)

- 10 Buchstaben zwischen L und n
- Prozess, bei dem die Übersetzung in den Quellcode integriert und die endgültige Ausgabe erstellt wird
- Zur Lokalisierung gehört auch die Anpassung von Inhalt wie z.B. das Datum und die Währung an ein Land

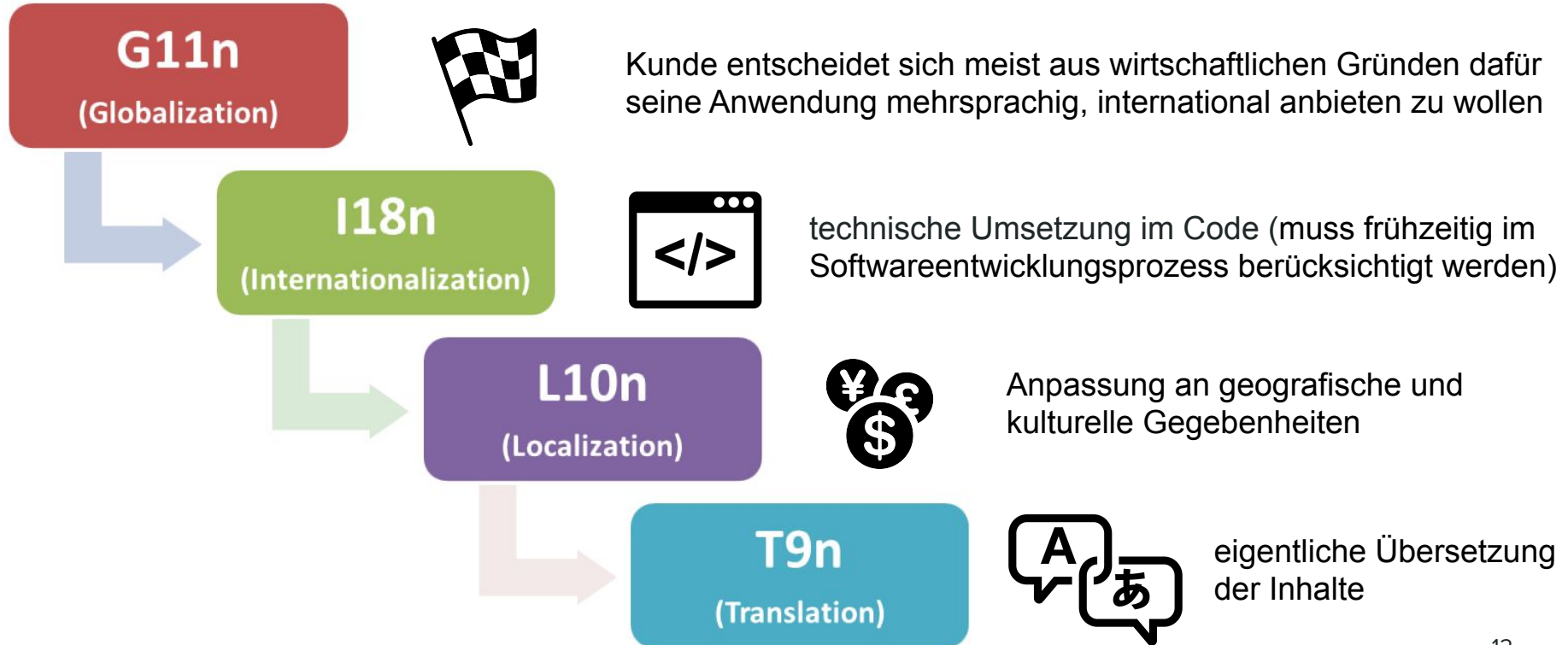
G11n, I18n, L10n und T9n



Translation (T9n)

- 9 Buchstaben zwischen T und n
- Prozess der Übersetzung von Text aus einer Sprache in eine andere Sprache
- In der Regel wird die Übersetzungsarbeit von:
 - professionellen Übersetzern
 - Agentur oder erfahrene freiberufliche Übersetzer
- oder
 - Macht der Gemeinschaft oder maschinelle Übersetzungen

G11n, I18n, L10n und T9n



ISO & Standardisierung

- internationale norm
- sprachen unabhängig
- global einsetzbar
- wettbewerbsfähig

Java

Java unterstützt den Entwurf internationalisierter Software durch:

- **Format**-Klassen erleichtern die Formatierung von Datum und Zahlen
- **ResourceBundle**-Objekte ermöglichen unterschiedliche Sprachen durch mehrstufige Übersetzungs-Dateien
- **Collator**-Klassen aus dem `java.text`-Paket können sprachabhängig sortieren

Die Steuerung erfolgt über *Locale*-Objekte oder die aktuelle Landeseinstellung der JVM (Clientseitig).

Java

lokalisierte Datumsangaben

java.text.DateFormat erzeugt String in landesspezifischen Darstellung

- gemäß der aktuellen Landeseinstellungen der JVM (`Locale.getDefault()`)
- oder expliziter Wahl einer Darstellung (z.B. `Locale.US`)

```
DateFormat df = DateFormat.getDateInstance( DateFormat.MEDIUM [, Locale] );  
System.out.println( df.format( new Date() ) );
```

en_US

Jul 4, 2019

en_GB

04-Jul-2019

de_DE

04.07.2019

Java Textersetzung durch Lokalisierungs-Dateien

Vorgehen:

1. alle Zeichenketten einer Landessprache durch symbolische Namen ersetzen
2. ResourceBundle-Objekte (Lokalisierungs-Dateien) im Projektordner anlegen
 - a. enthalten die Übersetzung von symbolischen Namen zu landessprachlichen Texten
 - b. können mehrstufig aufgebaut sein (DE vs. DE_DE / DE_AT / DE_CH)
3. Übersetzung der symbolischen Namen erfolgt zur Laufzeit

Besonderheit in Java: Passende Übersetzungen werden selbstständig gefunden, d.h. Lokalisierungs-Dateien müssen nicht vollständig sein.

Java Hierarchie der Lokalisierungs-Dateien

- bundleName_defaultLanguage
- bundleName_localeLanguage
- bundleName_localeLanguage_localeCountry
- bundleName_localeLanguage_localeCountry_localeVariant

```
# HelloWorld.properties  
Hello = Hello World.  
Bye = Bye.
```

```
# HelloWorld_de.properties  
Hello = Hallo Welt.  
Bye = Tschüss.
```

```
# HelloWorld_de_CH.properties  
Hello = Grüezi.
```



```
String baseName = "resources.HelloWorld";  
ResourceBundle bundle = ResourceBundle.getBundle( baseName [, Locale] );  
  
System.out.println( bundle.getString("Hello") );  
System.out.println( bundle.getString("Bye") );
```

Java

Live-Demo

HTML, XML & CSS

Warum einsetzen?

- Accessibility und Screenreader auf Webseiten
- Qualität der Suchergebnisse von Suchmaschinen (SEO)
- Verhalten einer Seite an Sprache anpassen
- Sprachspezifische Fonts laden
- Übersetzungstools unterstützen

Empfehlung: Sprachtags einbauen auch wenn es nicht wichtig erscheint.

Retrofitting ist sehr umständlich.

HTML, XML & CSS

Nützliche Language-Tags

HTML:

`<html lang="it">` Sprache der Seite ist Italienisch.

`<html dir="rtl">` Seite wird von rechts nach links dargestellt.

XML:

`<html lang = „no“ xml:lang="no">` Daten in im XML-Dokument sind Norwegisch.

`<a xml:lang="en" href="xyz" hreflang="de">Click for German</a>`

Verweist auf die Sprache von Daten in einem externen Kontext.

HTML, XML & CSS

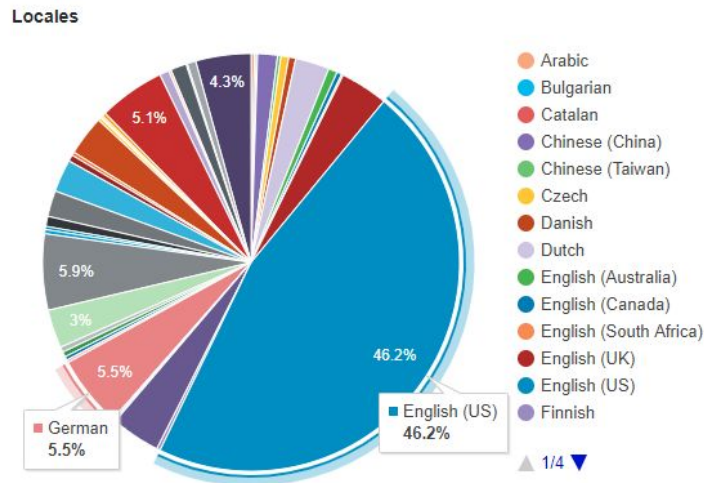
Nützliche Language-Tags

CSS:

```
body {  
    font-family: "Palatino Linotype", "Book Antiqua", Palatino, serif;  
}  
  
:lang(ar) {  
    font-family: "Traditional Arabic", "Al Bayan", serif;  
    direction: rtl;  
}
```

Ist die Sprache der Seite Arabisch, wird die **Richtung** geändert und eine andere **Schriftart** geladen.

WordPress & GNU



- **Problem:** Themes/Plugins werden international genutzt aber auf Englisch programmiert
- **gettext** ermöglicht das Internationalisieren von Themes/Plugins
- WordPress bietet eigene gettext-Funktionen mit denen man als Entwickler arbeiten kann
- Kulturabhängige Daten werden von WordPress selbst übersetzt

WordPress

Strings erfassen

```
<h3>Leave a Comment</h3>
```



```
<h3><?php echo __("Leave a Comment"); ?></h3>
```

- Formulare
- Buttons
- Überschriften
- Beschriftungen
- Links

Alternative Funktionen:

```
<?php _e(); ?> - Gibt den Text aus  
<?php _x(); ?> - Kommentar an Übersetzer übergeben  
<?php _n(); ?> - Singular/Plural
```

WordPress Textdomain definieren

Ein unique Identifier der WordPress hilft Übersetzungen zuzuordnen.

```
function my_theme_load_theme_textdomain() {  
    load_theme_textdomain( 'my-theme', get_template_directory() . '/languages' );  
}  
add_action( 'after_setup_theme', 'my_theme_load_theme_textdomain' );
```

Textdomain den Strings übergeben:

```
<h3><?php echo __( "Leave a Comment" ); ?></h3>
```



```
<h3><?php echo __( "Leave a Comment", "my-theme" ); ?></h3>
```

WordPress Sprachdatei erstellen

Tools zur Erstellung der Sprachdatei (Portable Object Template → *.pot)

- WP-CLI: *wp i18n make-pot path/to/my-theme*
- PoEdit

```
my-theme.pot

#: comment.php:123
msgid "Leave a Comment"
msgstr ""

#: comment.php:142
msgid "Send Comment"
msgstr ""

...
```

WordPress Übersetzen

Es entsteht eine *.po-Datei (Portable Object), die die übersetzen Strings enthält

my-theme.pot

```
#: comment.php:123
msgid "Leave a Comment"
msgstr ""

#: comment.php:142
msgid "Send Comment"
msgstr ""

...
```



my-theme-de_De.po

```
#: comment.php:123
msgid "Leave a Comment"
msgstr "Hinterlasse einen
Kommentar"

#: comment.php:142
msgid "Send Comment"
msgstr "Kommentar schicken"

...
```

- es entsteht auch eine *.mo-Datei (Machine Object) die Maschinencode enthält
- Umstellung der Sprache im WordPress Backend → Front-End ist ebenfalls übersetzt

Beispiel: JavaScript

- Vue - JavaScript Framework, u.a. für Single Page Application
- Einzelne Bausteine der Website werden aktualisiert und tauschen Daten aus
- Ziel: komplette Applikation in einer HTML-Seite
- Link zum Beispiel:
 - <https://www.freecodecamp.org/news/how-to-add-internationalization-to-a-vue-application-d9cfdcabb03b/>

Beispiel: JavaScript - Bilder



Welcome to Your Vue.js App

For a guide and recipes on how to configure / customize this project, check out the [vue-cli documentation](#).

Installed CLI Plugins

[babel](#) [eslint](#)

Essential Links

[Core Docs](#) [Forum](#) [Community Chat](#) [Twitter](#) [News](#)



Vue.js 앱에 오신 것을 환영합니다.

이 프로젝트를 구성 / 사용자 정의하는 방법에 대한 안내 및 요리법, 체크 아웃 [vue-cli documentation](#).

설치된 CLI 플러그인

[babel](#) [eslint](#)

필수 링크

[Core Docs](#) [Forum](#) [Community Chat](#) [Twitter](#) [News](#)

Beispiel: JavaScript - Codeausschnitte

```
<h1>{{ $t('welcomeMsg') }}</h1>
<p>
  {{ $t('guide') }}<br>
  {{ $t('checkout') }}
  <a href="https://cli.vuejs.org" target="_blank" rel="noopener">vue-cli documentation</a>.
</p>
<h3>{{ $t('plugins') }}</h3>
```

```
const messages = {
  'en': {
    welcomeMsg: 'Welcome to Your Vue.js App',
    guide: 'For a guide and recipes on how to configure / customize this project,',
    checkout: 'check out the',
    plugins: 'Installed CLI Plugins',
    links: 'Essential Links',
    ecosystem: 'Ecosystem'
  },
}
```

```
data() {
  return {
    languages: [
      { flag: 'us', language: 'en', title: 'English' },
      { flag: 'es', language: 'es', title: 'Español' },
      { flag: 'kr', language: 'kr', title: '한국어' },
    ]
  };
}
```

Beispiel: Nivea (Beispiel für Globalisierung)

Japan:

<https://www.nivea.co.jp/>

Deutschland:

<https://www.nivea.de/>

China:

<https://www.nivea.com.cn/>

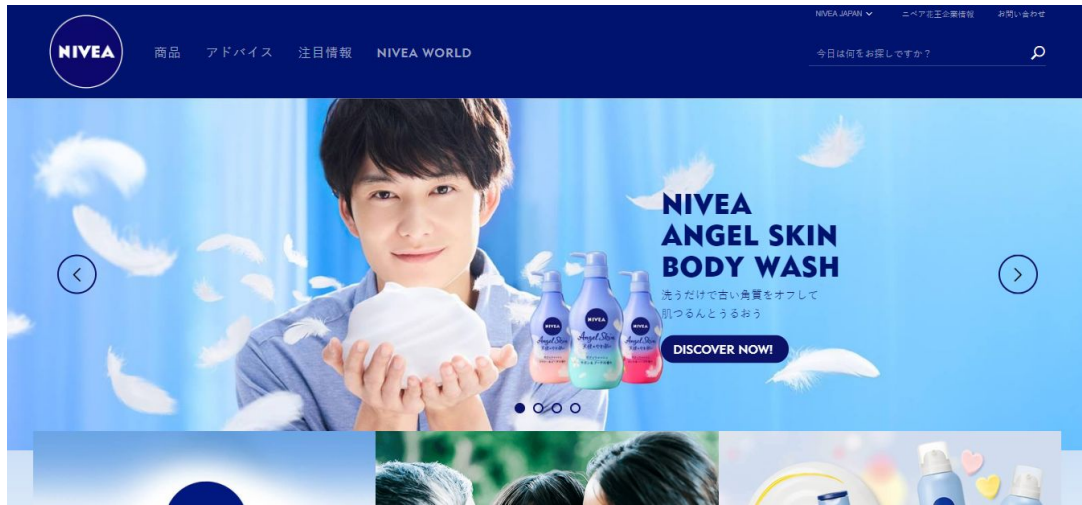
Kenya:

<https://www.nivea.co.ke/>

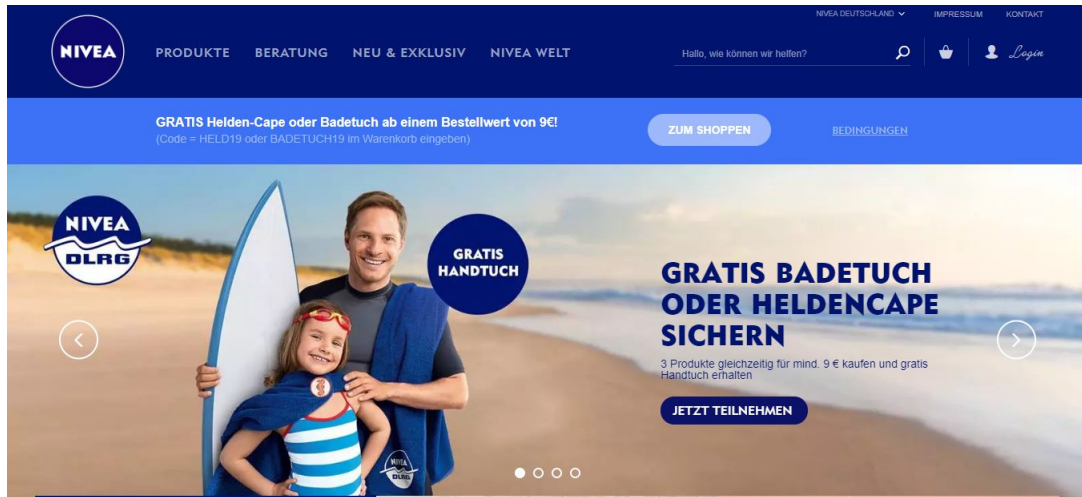
Zusammenfassung:

Die Startseite von Nivea wird durch den Inhalt, die Sprache und die Produkte die angeboten werden auf das Land angepasst.

Unterstützte Länder könnt ihr unter <https://www.nivea.com/> finden.



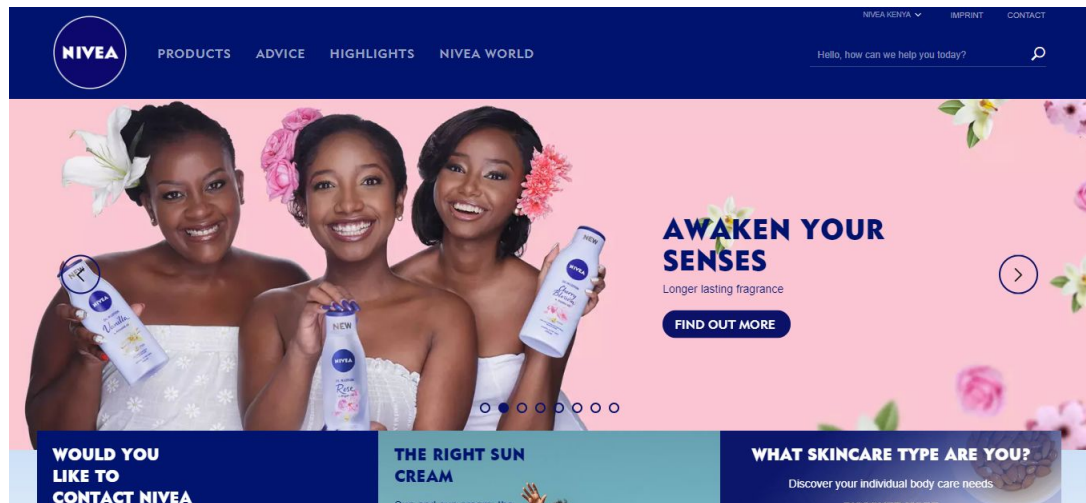
Japan



Deutschland



China



Kenya

Beispiel: McDonald's (Beispiel für Globalisierung)

Japan:

<http://www.mcdonalds.co.jp/>

Deutschland:

<https://www.mcdonalds.de/>

Zusammenfassung:

Die Startseite von McDonald's unterscheidet sich in ihrem Aufbau von Land zu Land.



Deutschland



Japan

Fazit

- globale wettbewerbsfähigkeit
- grösseres zielpublikum
- internationale zusammenarbeit
- grössere umsätze
- Mehraufwand durch Übersetzung
- unterschiedliche lokale gewohnheiten
- größerer verwaltungsaufwand
- größerer wettbewerb

Vielen Dank für die Aufmerksamkeit!

Quellen

G11n, I18n, L10n und T9n

- <https://www.w3.org/International/questions/qa-i18n.en.html>
- [https://de.wikipedia.org/wiki/Internationalisierung_\(Softwareentwicklung\)](https://de.wikipedia.org/wiki/Internationalisierung_(Softwareentwicklung))
- <https://aboutlocalization.wordpress.com/2016/01/19/globalization-vs-internationalization-vs-localization-vs-translation-what-you-need-to-know/>
- <https://www.linkedin.com/pulse/difference-between-g11n-i18n-t9n-l10n-satish-singh>
- <https://firmenblog.lionbridge.com/internationalisierung-von-websites/>

- <https://www.w3.org/International/questions/qa-lang-why>
- <https://www.w3.org/International/tutorials/language-decl/>
- <https://www.w3.org/International/questions/qa-when-xmllang>
- <https://www.gnu.org/software/gettext/>
- <https://developer.wordpress.org/themes/functionality/internationalization/>
- <https://developer.wordpress.org/themes/functionality/localization/>
- <https://wordpress.org/about/stats/>

Backup

Alle Systemsteuerung

Systemsteuerung > Alle Systemsteuerungselemente

Einstellungen des Computers anpassen

Akamai NetSession
Automatische Wiederherstellung
E-Mail (32-Bit)
Flash Player (32-Bit)
Infobereichssymbole
Maus
Region
Sprache
System
Wartungscenter
Windows-Mobilitäts

Region

Formate Aufenthaltsort Verwaltung

Format: Deutsch (Deutschland)

[Sortiermethode ändern](#)
[Spracheinstellungen](#)

Datums- und Uhrzeitformate

Datum (kurz):	TT.MM.JJJJ
Datum (lang):	TTTT, T. MMMM JJJJ
Uhrzeit (kurz):	HH:mm
Uhrzeit (lang):	HH:mm:ss
Erster Wochentag:	Montag

Beispiele

Datum (kurz):	02.07.2019
Datum (lang):	Dienstag, 2. Juli 2019
Uhrzeit (kurz):	23:07
Uhrzeit (lang):	23:07:28

[Weitere Einstellungen...](#)

OK Abbrechen Übernehmen

I18n-Demo - NetBeans IDE 8.0.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

<default config> Search (Ctrl+I)

I18nDemo.java

```
17  * @param args the command line arguments
18  */
19
20  public static void main(String[] args) {
21
22      // Windows Regionseinstellung sicher übernehmen (funktioniert seit Java 7)
23      Locale.setDefault( Locale.getDefault( Locale.Category.FORMAT ) );
24
25      // Importieren der Lokalisierungs-Datei (für die aktuelle Locale-Einstellung)
26      String baseName = "resources.HelloWorld";
27      ResourceBundle bundle = ResourceBundle.getBundle( baseName );
28
29      // Anlegen eines DateFormats (für die aktuelle Locale-Einstellung der
30      DateFormat df = DateFormat.getDateInstance( DateFormat.MEDIUM );
31
32      // Kontrollausgabe der aktuellen Ländereinstellung
33      System.out.println( Locale.getDefault() + "\n");
34
35      // Ausgaben
36      System.out.println( bundle.getString("Hello") );
37      System.out.println( df.format( new Date() ) );
38      System.out.println( bundle.getString("Bye") );
39      System.out.println();
40  }
41
```

main - Navigator

Members <em...>

I18nDemo

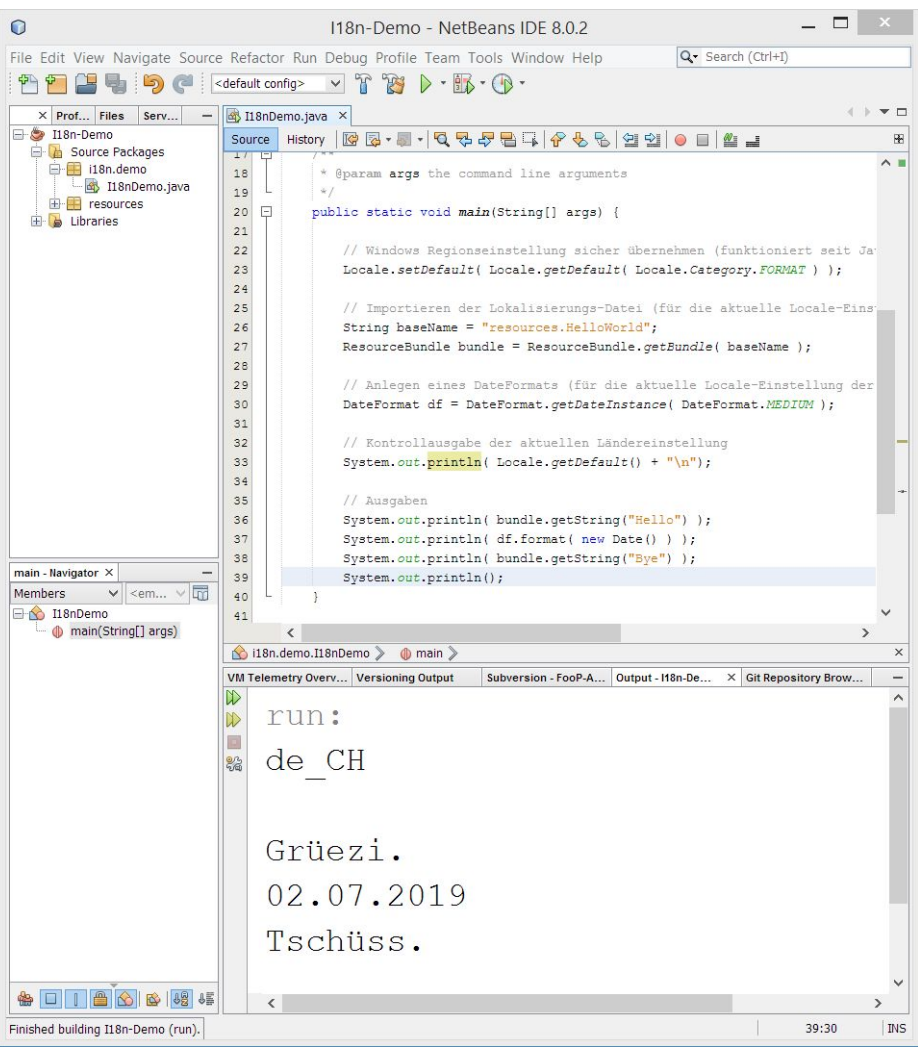
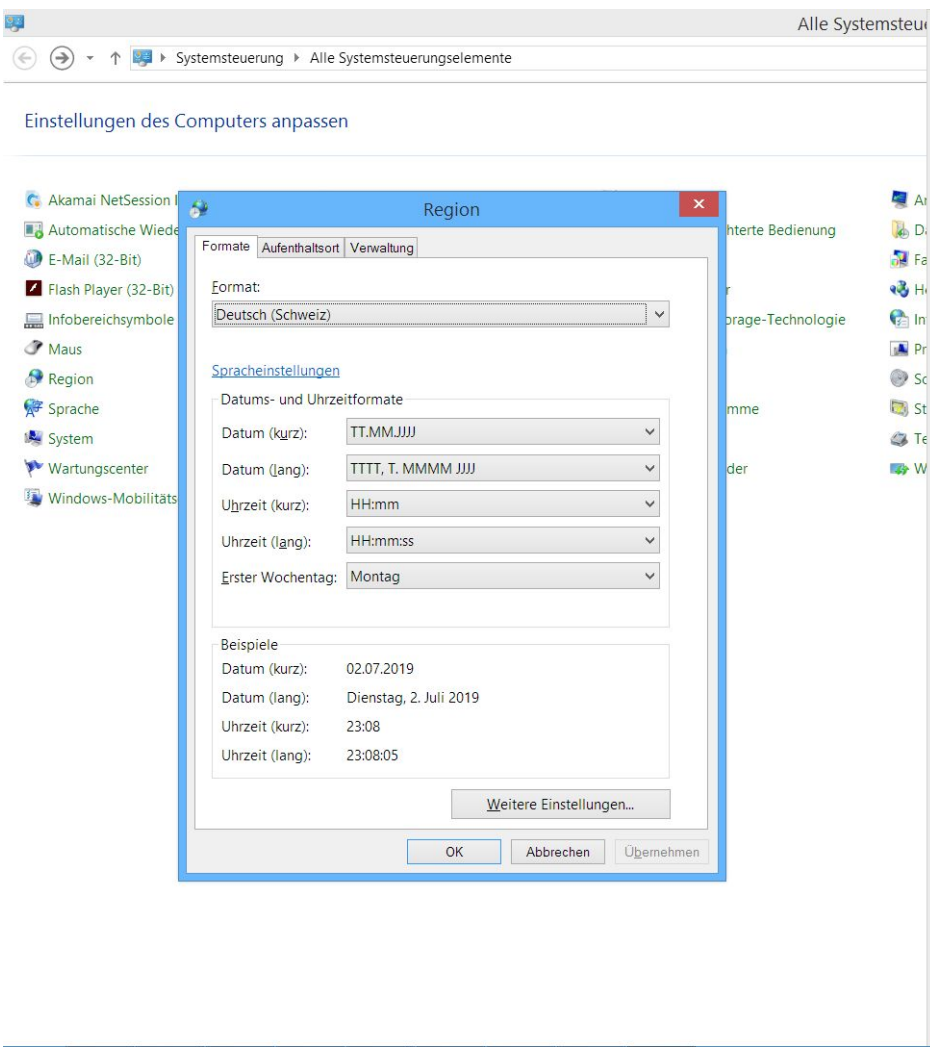
main(String[] args)

run:

de_DE

Hallo Welt.
02.07.2019
Tschüss.

39:30 INS



Akamai NetSession | Automatische Wiederherstellung | E-Mail (32-Bit) | Flash Player (32-Bit) | Infobereichssymbole | Maus | Region | Sprache | System | Wartungscenter | Windows-Mobilität

Region

Formate | Aufenthaltsort | Verwaltung

Format: Englisch (Großbritannien)

Spracheinstellungen

Datums- und Uhrzeitformate

Datum (kurz):	TT/MM/JJJJ
Datum (lang):	TT MMMM JJJJ
Uhrzeit (kurz):	HH:mm
Uhrzeit (lang):	HH:mm:ss
Erster Wochentag:	Monday

Beispiele

Datum (kurz):	02/07/2019
Datum (lang):	02 July 2019
Uhrzeit (kurz):	23:03
Uhrzeit (lang):	23:03:24

Weitere Einstellungen...

OK Abbrechen Übernehmen

I18n-Demo - NetBeans IDE 8.0.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+F)

I18n-Demo

- Source Packages
 - i18n.demo
 - I18nDemo.java
 - resources
 - Libraries

I18nDemo.java

```
17  * @param args the command line arguments
18  */
19
20  public static void main(String[] args) {
21
22      // Windows Regionseinstellung sicher übernehmen (funktioniert seit Java 7)
23      Locale.setDefault( Locale.getDefault( Locale.Category.FORMAT ) );
24
25      // Importieren der Lokalisierungs-Datei (für die aktuelle Locale-Einstellung)
26      String baseName = "resources.HelloWorld";
27      ResourceBundle bundle = ResourceBundle.getBundle( baseName );
28
29      // Anlegen eines DateFormats (für die aktuelle Locale-Einstellung der
30      DateFormat df = DateFormat.getDateInstance( DateFormat.MEDIUM );
31
32      // Kontrollausgabe der aktuellen Ländereinstellung
33      System.out.println( Locale.getDefault() + "\n");
34
35      // Ausgaben
36      System.out.println( bundle.getString("Hello") );
37      System.out.println( df.format( new Date() ) );
38      System.out.println( bundle.getString("Bye") );
39      System.out.println();
40  }
41
```

main - Navigator

- Members
 - <em...>
- I18n-Demo
 - main(String[] args)

run: en_GB

Hello World.
02-Jul-2019
Bye.

39:30 INS

