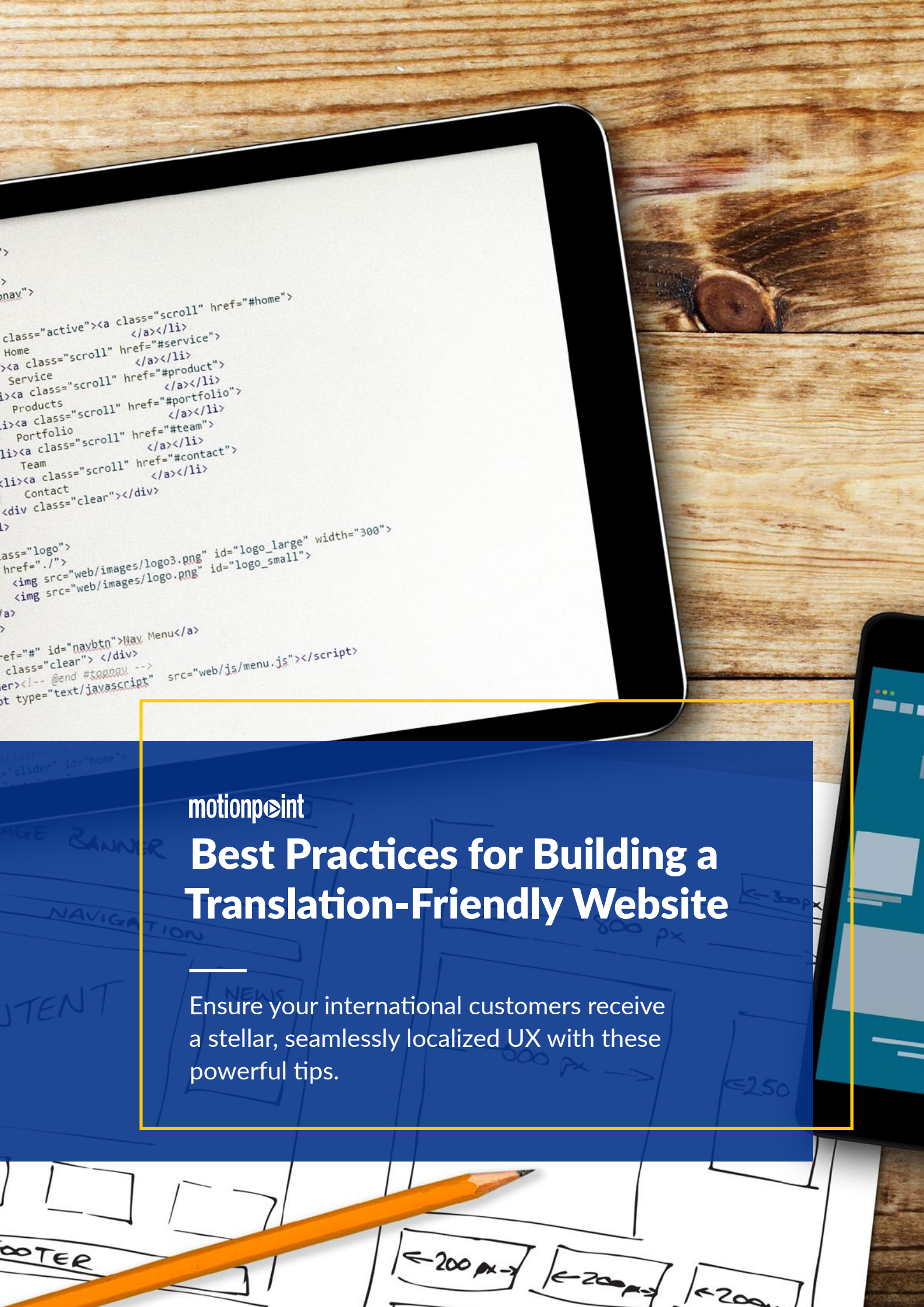


motionpoint

Best Practices for Building a Translation-Friendly Website

Ensure your international customers receive a stellar, seamlessly localized UX with these powerful tips.

Introduction

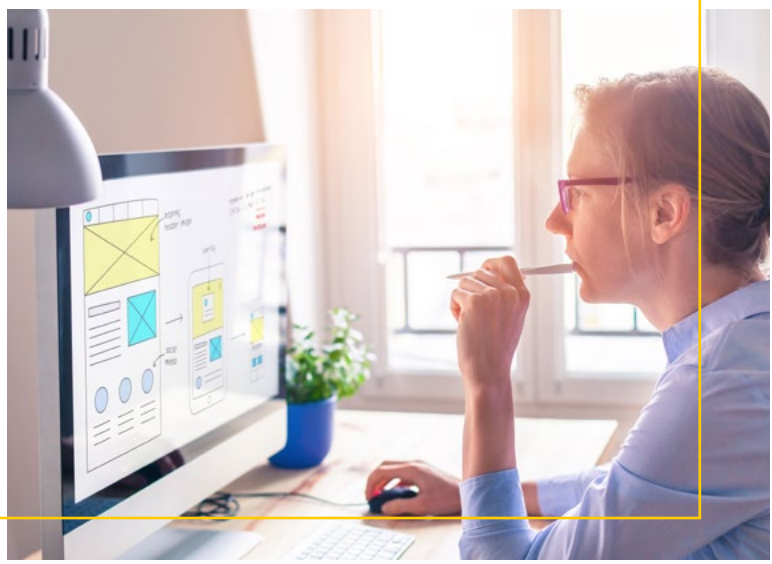
Creating an intuitive UX for international customers is essential to creating a positive online experience. The best way to achieve this is by **translating your website and digital content** into your customers' preferred languages.

But when a website isn't initially designed with localization in mind, international users can wind up experiencing partially-translated content, confusing or broken menu links, and other functionality problems.

Following the best practices in this document can help designers, developers and content creators optimize their websites for localization. You'll discover:

- ✓ Unique ways to design websites to accommodate the nuances of localization
- ✓ How to avoid potential functional disruptions and design problems
- ✓ Coding considerations for multimedia and omnichannel content

The issues that are created when designers don't follow these best practices can lead to customer confusion—and can even drive them away for good.



Some Context

Website translation enables you to publish content that is understandable to anyone who speaks the language, regardless of where they might live. This "region neutral" approach to word choice helps establish baseline credibility for entering new markets.

Website *localization* "levels up" your translations by incorporating market-specific cultural and linguistic nuances. These special translations often drive deeper engagement.

To accommodate both approaches, companies should enable their websites' back-end architectures to handle the special needs of translated sites, such as:

- Various languages

- Unique language character sets
- Currencies
- Site search capabilities and more

Failing to proactively prepare your origin website for a translation project often leads to broken functionality on the multilingual website. Pages may not be fully translated. Other issues—such as "broken" page designs created by longer translations—also arise.

To achieve optimum results, design and develop your origin site using these tips.

Design/Layout Considerations

1 Using HTML Markup & Text

a. Example: Use the css class attribute on a b or i element to identify why the element is being used.

b. Avoid using
 tags. These may create design issues down the road due to *word growth* (sometimes called *word expansion*), a common side effect of translation in which the translated content has more words or characters than the source-language content.

Depending on the language, translated content can "grow" by up to 30%.

2 Images

a. Externalize text from imagery. Benefits include:

i. Accelerates turnaround time for image translation, which is important for frequently changing imagery.

ii. Saves money by eliminating graphics work.

iii. Improves search engine rankings for terms originally contained within imagery.

b. Avoid using width and height attributes. The dimensions of the image should define its size on the page. Images should "float freely" on the page.

c. Avoid embedding prices or dynamic offers within imagery—unless via externalized text—to eliminate the risk of displaying stale pricing or offers.

d. Avoid merging images to create a larger image where text is used.

e. Avoid using absolute positioning for nested elements.

i. The "parent" element should set the positioning of the nested elements on the page. If an image needs to be resized, the new dimension should "push" the surrounding elements (text or other images) vertically or horizontally, rather than overlapping them.

ii. The positioning source code of the element should represent its visual positioning on the page. If an image is visually placed on the right side of the page, its positioning code should be align: right, float: right (positioning shouldn't be defined as distance from left).

f. Whenever possible, organize sprite images vertically (unless it dramatically affects file size), as this accommodates the word growth that naturally occurs when translating an image in multiple languages.

3 Alignment and Word Growth

a. Page templates should be fully dynamic, accommodating word growth or reduction.

b. Allow word wrap rather than word overflow (for text).

c. CSS Custom Identifier

i. Providing support for features that may not be used until localization occurs. For example, adding markup in your DTD to support bidirectional text, or for identifying languages. Or adding CSS support for vertical text or other non-Latin typographic features.

ii. Use web fonts, and include fonts that support multiple languages (such as Arial).

d. Separate Localizable Content/Code Element

i. Separating localizable elements from source code or content accommodates the ability to load or select localized alternatives, based on the user's international preferences.

e. The Arabic language changes content to be read from right to left, which often requires custom layout/ formatting from the origin website.

i. Use both global and localized CSS to align text without fixed values

ii. Localized CSS is useful for positioning floating and/or fixed elements

iii. Avoid in-line style, which is especially problematic for languages that read right to left

iv. Avoid controlling capitalization via CSS

f. Create “breathing space” on imagery that contains text (if you must embed text within imagery) and within JS, CSS and HTML templates to accommodate word growth.

g. If you must embed content within JS, name JS files in a way that makes it as clear as possible where the content appears on the website.

h. Approximate word and character growth per language (vs. English):

Language	Size (relative to English)
English	0%
German	+18%
Spanish	+20%
French	+20%
Portuguese	+17%
Italian	+17%
Russian	+18%
Arabic	+5%
Korean	-37%
Chinese	-11%
Japanese	-40%

Coding Considerations

Functionality – Display

a. Select custom fonts that offer support for alternate language character sets.

b. Don't use on-page literals to drive logic. Ensure content that is displayed in the page is not used for driving functionality or content.

c. Don't use URL literals to drive logic.

d. Use a JSON key naming convention, which allows easier identification of content to process for translation.

e. Avoid concatenating phrases from multiple dynamic sources (such as CMS, JSON, JS, XML). Each should stand alone to allow for proper translation.

f. Consolidate error messages into a single JS error message file.

Functionality – Backend

a. Consider that time, currency, dates, zip codes, etc. may follow different conventions in different international markets. These should be reflected on the localized site. To avoid additional work later on, these conventions should be supported by any user input validation on the origin site.

b. Allow users to change units of measure in fields (e.g., Imperial to metric, Fahrenheit to Celsius).

c. Leverage existing content (i.e., already translated content) for Facebook Open Graph (OG), alt tags and other metadata to avoid additional translation costs.

d. Beware the use of "fields" because how you concatenate them may not make sense in other languages. This requires additional QA and translation modifications down the road. It also delays time to market.

i. Break segments containing dynamic fields (e.g., "*Product Name+ is in stock").

e. Use JS f(x) to alphabetize lists and combo boxes, client-side.

f. Provide a stable pre-production environment to allow translation prior to production.

Videos

Build closed captioning into videos. This improves usability and accelerates the translation process. It also reduces the time investment required to create a translated version of the video. This best practice reduces costs.

SEO & Localization Considerations

Just like your origin site, you should have a solid SEO strategy for each of your translated sites. Following specific best practices for international SEO can improve a localized sites' visibility in regional search engines. Organic traffic and on-site engagement can increase, too.

International Domain Structure

a. Each translated version of your origin site should adopt its own domain structure (ccTLDs vs. subdomains vs. subdirectories). Discuss this decision with your marketing team, SEO agency and other stakeholders. Don't use URL parameters, cookies or scripts to deploy your translated site structures.

Reference Link: https://support.google.com/webmasters/answer/182192?hl=en

b. Explicitly notify search engines about your multiple translated sites by including *hreflang* tags on those sites. This ensures regional search engines display the correct localized version of your site to the right customers. Hreflang attributes can be inserted in a few different ways:

i. HTML (<head>)

ii. HTTP headers

iii. Sitemaps

Tracking KPIs

Track and gauge each of your translated sites independently. This entails creating different profiles in Google Search Console (there are alternate solutions based on your target market, such as Baidu Webmaster Tools for China) and Google Analytics (Adobe Analytics, etc.).

On-Page SEO Content

a. Technical SEO, just as with your origin website, should be implemented by fluent SEO specialists. This ensures translated sites can be optimized for crawlability and

indexability appropriately. This includes, but is not limited to:

i. Translating metadata & structured data

ii. Understanding the search engine market based on the region (such as China's Baidu, Russia's Yandex and South Korea's Naver)

iii. Implementing *hreflang* tags

iv. Translating URLs

Customized Localization

a. Identify areas of the site that should contain unique customized translated content (for different geographies, languages, or visitor types).

b. Determine which of the following ways to localize content should be applied in each instance:

i. **Back-end Dynamic Customization:** This is served by your CMS/testing/ personalization engine typically triggered via a cookie, IP address, session variable or URL parameter.

ii. **Front-end Custom Pages:** Often used for static customizations for certain languages or countries (e.g., terms & conditions, which vary by country).

Recognizing when internationalization practices should be implemented is the first step in ensuring your team and budget are used more efficiently during a localization project. Proactively investing the effort to properly prep your site for translation will prevent needless costs and restructuring down the road.

About MotionPoint

MotionPoint solves the operational complexity and cost of localizing web and digital content. Unlike all other approaches, our technology and turn-key solution are built specifically for this purpose.

We translate, deploy, and operate multilingual websites and other digital content, optimizing the customer experience across all channels.

motionpoint

MotionPoint Corporation

info@motionpoint.com

www.motionpoint.com