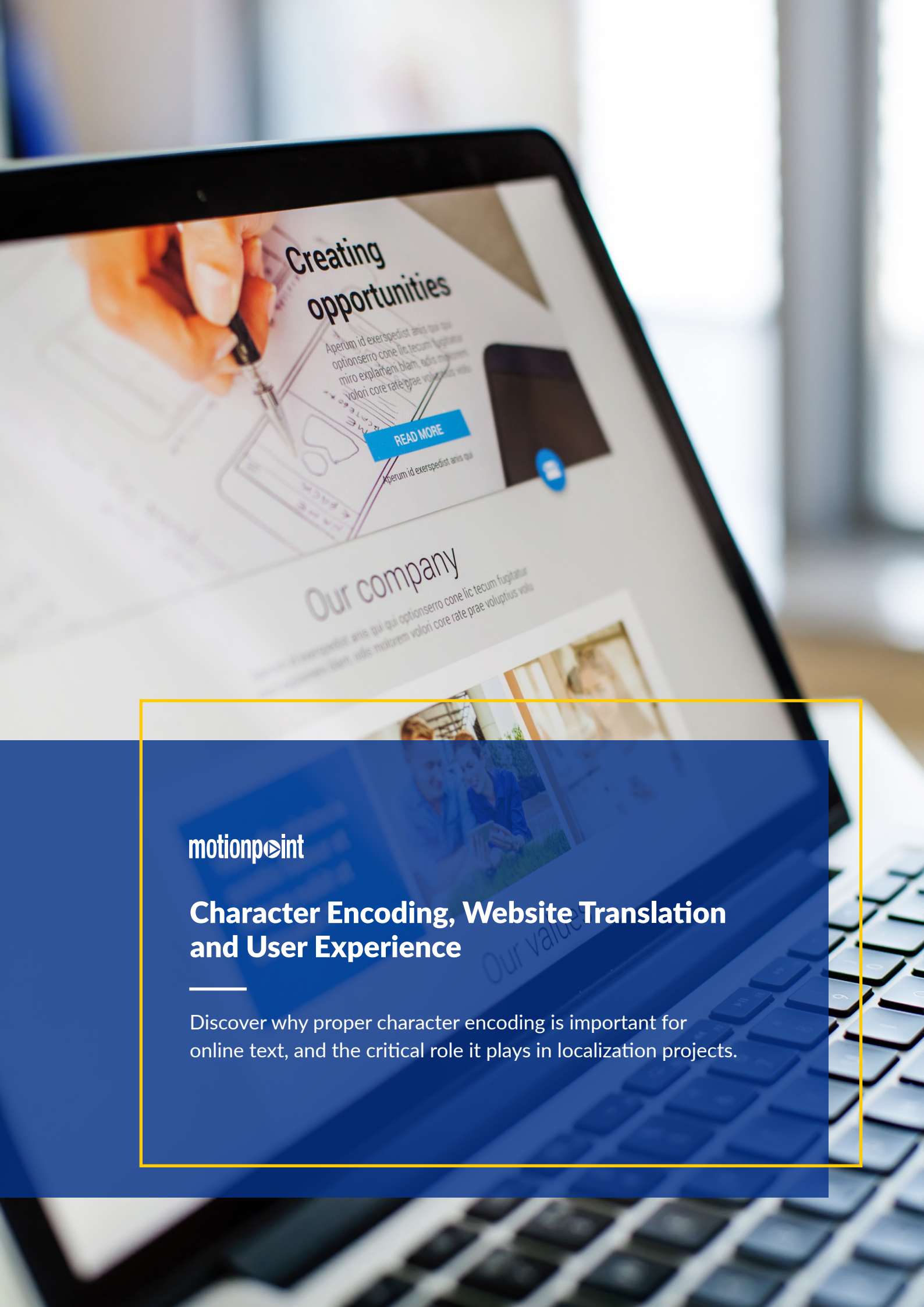


motionpoint

Character Encoding, Website Translation and User Experience

Discover why proper character encoding is important for online text, and the critical role it plays in localization projects.

Introduction

When online digital content is translated from one language to another, an unfortunate—and common—side effect can occur when this translated content is transported to a different medium.

Simple sentences that contain accented letters or special formatting can appear malformed when copied from one file to another. Specific characters and punctuation elements usually render as a series of question marks or random non-standard characters.

Why is this happening?

Character Encoding

Character encoding is a way of telling a computer how to interpret digital data into letters, numbers and symbols. This is done by assigning a specific numeric value to a letter, number or symbol. There are a number of character encoding sets in use today, but the most common formats in use on the World Wide Web are ASCII, UTF-8 and Unicode.

In order to properly render translated digital content, the correct character set (aka character encoding) must be used. Here's some history on character sets, along with some tips on how to properly leverage them for your website translation projects.

ASCII

In 1963, the ASCII (American Standard Code for Information Interchange) character encoding scheme was established as a common code used to represent English characters, with each letter assigned a numeric value from 0 to 127.

Most modern character encoding subsets are based on the ASCII character encoding scheme, and support several additional characters.

ANSI/Windows-1252

When the Windows operating system emerged, a new standard was quickly adopted known as the ANSI character set. The phrase "ANSI" was also known as the Windows code pages (Code Page 1252),

even though it had nothing to do with the American National Standards Institute.

Windows-1252 or CP-1252 (code page 1252) character encoding became popular with the advent of Microsoft Windows, but was eventually superseded when Unicode was implemented within Windows.

ISO-8859-1

The ISO-8859-1 character encoding set features all the characters of Windows-1252, including an extended subset of punctuation and business symbols. This standard was easily transportable across multiple word processors, and even newly released versions of HTML 4.

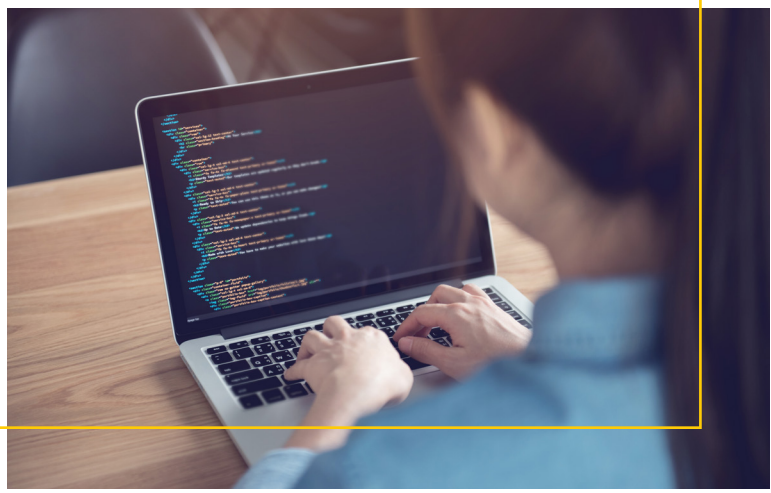
ISO-8859-1 was a direct extension of the ASCII character set. While support was extensive for its time, the format was still limited.

UTF-8

After the debut of ISO-8859-1, the Unicode Consortium regrouped to develop more universal standards for transportable character encoding.

UTF-8 is now the most widely used character encoding format on the web. UTF-8 was declared mandatory for website content by the Web Hypertext Application Technology Working Group, a community of people interested in evolving the HTML standard and related technologies.

UTF-8 was designed for full backward compatibility with ASCII.



The Big Idea

So it's clear that each character set uses a unique table of identification codes to present a specific character to a user. If you were using the ISO-8859-1 character set to edit a document and then saved that document as a UTF-8 encoded document without declaring that the content was UTF-8, special characters and business symbols will render unreadable.

Most modern web browsers support legacy character encodings, so a website can

contain pages encoded in ISO-8859-1, or Windows-1252, or any other type of encoding. The browser *should* properly render those characters based on the character encoding format not being reported by the server.

However, if the character set is not correctly declared at the time the page is rendered, the web server's default is usually to fall back without any specific character encoding format (usually ASCII).

This forces your browser or mobile device to determine the page's proper type of character encoding. Based on the WHATWG specifications adopted by W3C, the most typical default fallback is UTF-8. However, some browsers will fall back to ASCII.

Tips and Next Steps

To ensure your users are seeing the correct content on your HTML production pages, **be sure the content is saved in and encoded using UTF-8.** Declare the encoding type within your page with the use of metatags.

Also make sure your server is delivering the correct data. Even if the data on your page is correctly encoded in UTF-8 and declared on the page, your server may be serving up the page with an HTTP header that is read by the end user as a different encoding. **Ensure the HTTP Content-Type header has UTF-8 specified as the encoding type.**

Following these specifications will easily facilitate website translation into various languages without having the need to decode and re-encode into other character encodings across the multichannel media that's used on the web today.

About MotionPoint

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

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

motionpoint

MotionPoint Corporation

info@motionpoint.com

www.motionpoint.com