1 Introduction

This introductory chapter is divided into seven sections, covering the overall context for this book, some justifications as to why a new book is required on the topic of localization, a brief explanation of the key terminology used, the intended audience, an overview of the book's structure, the scope of the book, and the conventions used throughout this book.

1.1 Context for this book

Desktop computers were introduced in the 1980s and during that decade hardware manufacturers and software publishers realized that in order to sell their products in other markets or countries, they would need to adapt them so that they would still be functional in different environments. This adaptation became known as *localization* since target countries or groups of countries were also referred to as *locales*. Localization was required because computers at the time relied on very different character sets so a program written in, say, Spanish and encoded in a Western encoding would not run properly on a Japanese operating system. Since then, localization processes have become more sophisticated and are often coupled with internationalization processes, which aim at preparing a product for localization. Note that, because of the length of the words internationalization and localization, the words are commonly shortened to *i18n* and *l10n*. These acronyms are 'quoting the first and last letter of each word, and replacing the run of intermediate letters by a number merely telling how many such letters there are.'[1]Adapting a product to a specific market or locale is of course not specific to the Information Technology (IT) industry, as any business hoping to operate successfully on a global scale is likely to rely on transformation processes so that their equipment, medicines or food products meet, or even exceed local regulations, customs and expectations. In this book, however, the focus is on software applications, which are also referred to as applications or *apps*.

1.1.1 Everything is an app

To an end-user, an app is often limited to the visual interface they use to accomplish a specific task. Depending on the complexity of the task(s) an app

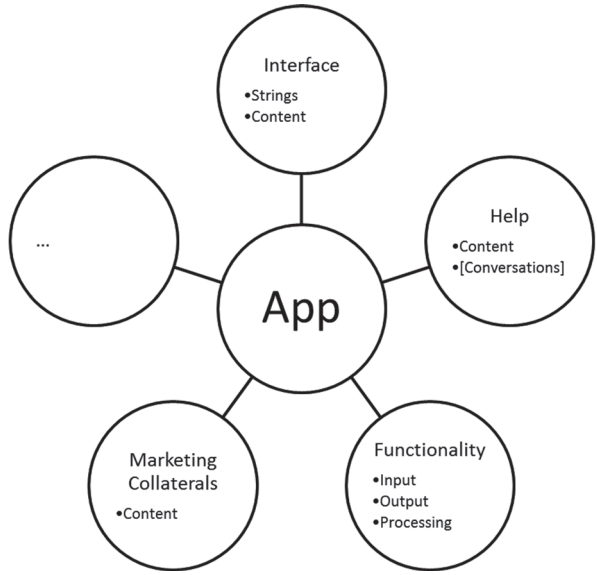

Figure 1.1 Components of a software application's ecosystem

is supposed to perform, additional components may become apparent over time. Such components can be referred to as an application's *digital ecosystem*, a non-exhaustive example of which is presented in Figure 1.1.

Most of these components should be familiar to anybody who has ever come across a software application in their life (be it a desktop application, a mobile application or a Web-based application). An application is obviously equipped with an interface, which is composed of textual strings (e.g. menu items) and content (e.g. informative content such as news items or pictures). While most app users probably know that help content is also available, finding and using such content is not as frequent as using the actual application's functionality (which is why help or support often takes place through online or physical conversations). An application's functionality often relies on user input which must be processed using procedures or algorithms in order to generate some output. Depending on the type of application, some marketing, training and sales-related content may also be generated, but this may not be as relevant from an end user's perspective. Figure 1.1 shows, however, that the ecosystem surrounding an application can become quite large if the application turns out to be a success. As far as this book is concerned, the focus is placed on those components that make apps different from other content types (e.g. a perfume's marketing brochure or a drugs information leaflet), thus requiring specific processes such as localization.

While software localization emerged in the IT sector, it is now prevalent in other sectors, especially those that have an online presence, be it through Web sites (which are very often indistinguishable from Web applications) or Web services. Any online digital content that is generated by online systems

or apps, can now be subject to some form of localization in order to reach as many users as possible. In this sense, no conceptual distinction is made in this book between Web sites, mobile apps or desktop programs: all of these are *apps*, whose digital ecosystem may vary in size depending on its user base. It should be stressed that contributions to this digital ecosystem need not solely originate from an authoritative app developer or publisher. App users are now increasingly directly and indirectly taking part in various aspects of an application's lifecycle, ranging from funding and suggesting features to testing and writing reviews. For instance, Kohavi et al. (2009: 177) explain that 'software organizations shipping classical software developed a culture where features [were] completely designed prior to implementation. In a Web world, we can integrate customer feedback directly through prototypes and experimentation.' With the Web 2.0 paradigm, publishing cycles have also been dramatically reduced, thanks to easy-to-use online services including collaboration tools. These tools and services have democratized the content creation process, which in turn has had an impact on localization-related processes.

1.1.2 The language challenge

Users are spending more and more time online, thus requiring content to be available in a language they can understand. A recent *User language preferences* survey conducted upon the request of the European Commission's Directorate-General for Information Society and Media (Gallup Organization 2011: 7) found that 'a great majority of Internet users in the EU used the Internet on a daily basis in the past four weeks: 54 per cent said they had gone online several times a day in that time frame and 30 per cent said it had been about once a day.' Such figures suggest that online opportunities exist for those companies that are able to reach users, despite potential language barriers. The area of Web localization had already been perceived as the 'fastest-growing area in the translation sector' more than a decade ago by O'Hagan and Ashworth (2002: xi) and it has never been more relevant than today. This is no surprise considering the ever-increasing amount of content to be translated in a very limited period of time. Even though localization does not only involve translation, publishers are often striving for a simultaneous publication of their information in multiple languages.

As far as multilingual Web sites are concerned, Esselink (2001: 17) warns that 'the frequency of updates has raised the challenge of keeping all language versions in sync (…), requiring an extremely quick turnaround time for translations.' However, providing information before it becomes obsolete is sometimes not possible for publishers, and some content is published exclusively in the language in which it has been authored. Yunker (2003: 75) remarks that 'unless the target audience consists of only bilinguals, this approach is bound to leave people feeling left out.' The lack of global distribution and accessibility has been highlighted by Pym (2004: 91) and is reflected by three types of locales: the *participative* locale consists of users who are able to access information in a language they can understand. These users are then able to act upon the information they have

accessed. The *observational* locale consists of users for whom it is too late to do anything with the information they access. They are able to access it in their own language, but by the time this information is translated, it is obsolete. The *excluded* locale consists of users who are never given the chance to gain access to information in a language they can understand.

Giammarresi (2011: 17) mentions that there are two main reasons for a company to localize its products: a reactive approach 'if one international customer has expressed interest in purchasing a localized version of one of the company's products' or a strategic approach 'if the company has decided to expand into one or more new international markets'. While this may be the case in certain scenarios, two other reasons should also be mentioned: the interest of users (not necessarily customers) to help localize the product (from an altruistic perspective) and the laws and regulations that are in operation in certain countries. These four main factors driving localization-related activities (global user experience, revenue generation, altruism and regulations) are discussed next.

1.1.3 The need for localization

The first factor concerns the user experience. For instance, the *User language preferences* survey mentioned in the previous section found that while some users feel comfortable reading or watching Web content using a language which is different from their native language, a majority of users expect to be able to interact with content (search, write, manipulate) in the language of their choice (e.g. majority of Europeans). A slim majority (55 per cent) of Internet users in the EU said that they used at least one language other than their own to read or watch content on the Internet, while 44 per cent said that they only used their own language. These numbers are more or less aligned with the survey carried out by the International Data Corporation in 2000 within the framework of the Atlas II project. Based on the results obtained from 29,000 Web users, they had estimated that by 2003, 50 per cent of Web users in Europe would be likely to favour sites in their native language (Myerson 2001: 14). These findings suggest that in order to provide a truly comfortable user experience, Web sites should offer some language support, which may involve some form of content localization (and possibly internationalization). Regardless of the reasons for not fully localizing online content (time, cost, lack of resources), the consequences of having content that is only partly localized should not be underestimated.

The second factor is revenue generation. A Common Sense Advisory report found that a major driver for any corporate involvement in global markets is always new revenue and market share opportunities.[2] DePalma et al. (2011: 2) report that 'high-tech hardware and equipment makers generate more than a quarter (27.1 per cent) of their revenue account from global markets [and] oil and gas companies earn 23.6 per cent of their income outside the United States.' In order to be able to compete in these markets, however, companies often have to break the language barrier, and localize some of their content, products or services. Since this requires an upfront investment, these companies must have

some level of confidence that this investment will be justified, or that they will get some Return On their Investment (ROI). According to Zouncourides-Lull (2011: 81), it is therefore common in localization projects to calculate costs 'using parametric estimation, by applying standard rates (e.g. cost per word) and possible revenue'. The pervasive use of translation memory technology has greatly influenced this approach, by providing a quick way to calculate how much it would cost to translate new or legacy content. When these ROI calculations fail to convince executive sponsors, or when the prospect of having to manage and support a number of languages is too daunting, language barriers remain, and opportunities are lost. This is particularly visible with smaller companies, which do not necessarily have the budgets or expertise to embrace localization. For instance, a recent nationwide survey of Irish hotels has found that only 18 per cent of those sampled offer languages other than English on their Web site.[3]

The third factor is altruism. When the opportunities described earlier are lost, volunteers sometimes decide to contribute some of their time to localize content. This is especially visible in the IT sector with open-source projects such as LibreOffice.[4] Altruism also applies to Non-Governmental Organizations (NGO) who rely on motivated volunteer translators. A recent survey conducted by O'Brien and Schäler (2010: 9) found that 'support for [The Rosetta Foundation]'s cause and opportunities to increase professional experience emerged as the two greatest motivating factors'. Also, volunteer-based collaborative translation or crowdsourcing is becoming common in large for-profit corporations. This is especially the case when the product does not necessarily need to be released in a timely fashion or when the ROI for localizing the product is not convincing (but a lot of enthusiastic users are willing to contribute to the localization effort).

Finally, local laws play a very important role in determining whether and how content should be translated or localized, including, but not limited to, language laws, data protection laws and certification laws. In terms of European language laws, for example, the Toubon law in France, whose full name is Law 94-665 of 4 August 1994 (Article 2), mandates the usage of the French language when a product or service is presented, offered or described (e.g. in its user manual or terms and conditions).[5] In Ireland, the Official Languages Act of 2003 provides a number of legal rights to Irish citizens with regard to their interactions with public bodies using the Irish language.[6]

Data protection laws also play a very important role when it comes to the handling of digital content. For instance, in Germany, the Federal Data Protection Act (Bundesdatenschutzgesetz or BDSG) of 1 January 1978 prohibits the collection, processing and use of personal data, unless it is explicitly permitted by law or approved, usually in writing, by the person concerned.[7] This means applications must be localized appropriately from a functionality and location perspective when adapted for the German market.

Finally certification laws can also have an impact on the localization of an application. For example, the US Department of Commerce mentions that before being sold in China, software products need to be registered at the China Software Industry Association, and the registration approved by the Ministry of

Information and Industry. Besides, American firms cannot register their product directly since registration must be made by a Chinese entity.[8] This process is even more stringent for the sale of enterprise encryption software since it needs to comply with the Commercial Cryptography Administration Regulation.[9] This example shows that local customs and regulations (including testing and inspection procedures) can increase the complexity of a localization project. While this type of example will not be treated in detail, it is a good reminder that localization is more than just translation.

At the time of writing, the localization industry is also influenced by a number of mega trends, including the increasing popularity of mobile platforms. These trends are creating new challenges, which are changing the way traditional localization is currently being done. Some of these challenges, 'volume, access and personalization,' were identified in Van Genabith (2009: 4) and are briefly reviewed in the next section.

1.1.4 New challenges affecting the localization industry

The *volume* challenge is caused by the large amount of content being created online on a daily basis, not necessarily exclusively by an official application's publishers, but by a number of actors interacting with the application's digital ecosystem. This volume challenge is exacerbated by the velocity at which this content is created or updated. Whereas application publishing cycles used to be regular (involving substantial changes between two versions), content updates tend to be more incremental, thus leading to a more continuous and prioritized approach to localization. An example of prioritized localization is reported by Airbnb's Jason Katz-Brown, who acknowledges that Airbnb's 'websites and mobile apps have 400,000 words of English content [, so they] couldn't translate all of it to Japanese in just a few days. It was important to prioritize so that the most visible webpages, email templates, and core flows of the site were delightfully localized before launch.'[10]

The *personalization* challenge refers mainly to monolingual content processing (i.e. content may be adapted or personalized for a given user depending on their level of expertise rather their linguistic preferences or expectations). The *access* challenge, which was touched on earlier in this chapter, is characterized by how people get access to and interact with online digital content, increasingly using mobile devices. For instance, this was exemplified by the fact that more Apple iPads than Hewlett Packard PCs were sold in the last quarter of 2011.[11] This was also exemplified by the increase of worldwide sales of smartphones (around 250 million units sold in the third quarter of 2013, up 45.8 per cent from same quarter the year before).[12] Obviously sales increases are not consistent across all world regions, the increases being the most prominent in Asia/Pacific. This online digital content used to be referred to as Web content, but with the advent of mobile applications (or apps), the view that 'we can no longer make a clear distinction between software and content when we discuss localization' (Esselink 2003b: 6) is perhaps more valid now than it was ten years ago. This means current

localization processes must be re-visited, taking into account the impact such changes have on the actual translation process. Related to this challenge is the fact that more and more interactions with devices (e.g. computers or mobile phones) are increasingly using non-textual methods. While it was common-place for consumer software applications to be accompanied by printed manuals in the 1990s, these have been largely replaced by electronic formats (e.g. HTML or PDF) in the last decade. At the time of writing, however, it is no longer clear whether these text-based electronic formats will still be dominant by the end of the 2010s. Recent advances in natural language processing (including speech recognition and speech synthesis) have allowed speech-based applications to gain in popularity (e.g. Apple's Siri or Google's Voice Actions on the Android platform). From a localization perspective, some of these applications require new processes, since it is no longer sufficient to simply translate computer strings in order to help an end-user use an application or read digital content. Rather, the application must be equipped with the (local) resources (such as text, speech and graphic) that will allow end-users to interact with content in an effective manner.

Apart from these three core challenges, other challenges exist, such as the way the translation process is being conducted. The concept of collaborative translation and localization is not new since it has been used effectively for a number of years in IT open-source projects such as Mozilla or Linux (Souphavanh and Karoonboonyanan 2005) or not-for-profit projects (e.g. Wikipedia). However, it is now gaining popularity in corporate for-profit environments, as exemplified by Twitter's Translation Center.[13] Collaborative translation is sometimes difficult to distinguish from crowdsourced translation, which tends to rely on paid translations (for small fees) rather than free translations. However, the people paid using this approach may not all be professional, certified translators. This presents both a challenge and an opportunity for professional translators. The challenge is that work which would have been performed in a professional capacity a few years ago can now be done faster and possibly cheaper by a number of bilingual amateurs or hobbyists. The opportunity is that the quality of these translations cannot be guaranteed, so reviewing or management expertise may be sought from translators. Besides, it is unlikely hobbyists will respond well to strict deadlines, so jobs requiring well-defined turnaround times will still be allocated to professional translators.

Finally, machine translation (MT) is becoming more and more mainstream in the localization industry. In the 1990s translation memory became a *de facto* technology, and during the 2000s globalization management systems (GMS) gained in popularity. Over the last five years or so, the quality of (online) machine translation systems has improved dramatically (mainly driven by the progress in statistical machine translation). This has led individuals and organizations to rely on such technology to provide (basic) localized content in specific scenarios. One of the challenges for today's and tomorrow's translators is to come to terms with such technology, and be aware of the customization opportunities that can be brought to such systems to raise the quality bar further. The use of MT is of

course changing the translation process in a more dramatic way than translation memory changed it 20 years ago. Nowadays a lot of digital content tends to be pre-translated (using machine translation) with a view to being subsequently post-edited by translators. Since the post-editing process is sometimes seen as boring or tedious, a clear opportunity for translators is to become more technical and gain expertise in areas such as text processing (i.e. the manipulation of textual data using programmatic means) in order to have more upstream control on the pre-translation process. By understanding better what can be achieved through automation, it is the author's belief that translators can then focus on what they like doing best: translating, or being involved in a linguistic activity (such as translation quality assurance, translation memory maintenance or MT optimization). For this reason, this book will have a quite technical focus so that readers can gain an insight into a number of text processing techniques. One of the objectives of this book is to equip its readers with knowledge that may not be necessary to perform the translation task *per se*, but that can provide added value in specific circumstances.

1.2 Why a new book on this topic?

The challenges described in the previous section require new insights and solutions, some of which have only started to emerge. At the beginning of the twenty-first century, seminal books on localization were published, including Esselink (2000). While this book was relevant at the time, some of its content is no longer up-to-date, since (i) it had a strong bias towards Microsoft Windows, which no longer reflects the diversity of platforms used in today's heterogeneous app-focused world, and (ii) localization processes and strategies have changed dramatically. For example, the 'printed documentation' (Esselink 2000: 12) that used to accompany a boxed software application is now a distant memory as most software applications are now made available via digital downloads, which may be reached using physical cards. When software applications are still distributed in physical boxes (on a CD, DVD), the documentation is often included as a PDF file or set of files that Esselink (2000: 12) described as 'online help' (where *online* meant digital). This content, which contains guidance or reference materials, is usually accessed from the application itself by triggering a specific command (e.g. clicking a *Help* button). In recent times, however, this type of content has become harder to distinguish from the actual *online* content that may be made available by the application publisher via a Web site (e.g. technical support content). For instance, the latest versions of Microsoft Office offer users the ability to search for information using multiple content sources, including the default documentation content present on the user's hard disk as well as online repositories.[14]

Another important volume was Savourel (2001), which focused on the eXtensible Markup Language (XML) format for a technical audience, rather than focusing on readers with a background in translation studies. More recently, an edited volume on localization project management was published (Dunne and Dunne 2011), but it did not cover the translation and text processing techniques

used during the localization process. A recent 'interdisciplinary overview of web localization' can also be found in (Jiménez-Crespo 2013: 1), but the main target audience is 'students or scholars interested in doing research in this field', rather than (prospective) professional practitioners. There is therefore a need for a new book on the internalization and localization of apps and their digital ecosystem, focusing on new developments, such as mobile devices, and addressing these new challenges that are impacting the localization industry.

1.3 Conceptual framework and key terminology

A common question is whether another term besides translation is required. Indeed, why is localization different from translation? The term localization is often used to describe a process that encompasses more than the translation of a simple digital document (say, a Microsoft Word document) using a Computer Assisted Translation (CAT) tool such as Wordfast.[15] Other linguistic (and non-linguistic) activities are often, if not always, required to adapt an application to the needs of people whose main language is not the same as the one in which the product or service was originally developed. In order to make the translation process easier, cheaper, and to some extent more sustainable, some upstream processes are sometimes required to prepare source content. This set of processes is often referred to as internationalization, especially in the IT industry, where software must ideally be internationalized before it gets localized. Without this process, localization would still be possible, but the associated cost and effort would increase. Some downstream processes are also often necessary, such as a linguistic quality assurance process, to ensure that the translated product or service has not been impacted negatively by the translation process. Esselink (2003a: 69) also explains that localization differs from translation because of the nature of the activities (e.g. multilingual project management, terminology management, software testing), the technology used (e.g. software translation tools, CAT or MT) and the complexity of the projects (large projects with multiple file formats). Since all of these activities are related, they are often encompassed under a more generic term, *globalization*. Figure 1.2 shows a possible way to represent this conceptual framework, which is going to be used in the remainder of this volume.

As shown in Figure 1.2, localization activities are separated into three categories. The first category concerns the translation activities that can be effectively enhanced and supported by a myriad of translation tools (such as translation memory and machine translation). These activities mostly focus on the translation of textual content, such as user interface strings and user assistance content. The second category relates to non-translation activities, such as file processing and testing, which are necessary to *glue* the output of the translation activities into target files. The third category, which is labelled as *adaptation* in Figure 1.2, concerns activities that do not belong to the other two categories. Adaptation activities include the localization of non-textual content, such as graphics or videos, and the translation of content that requires a very high level of transformation, possibly

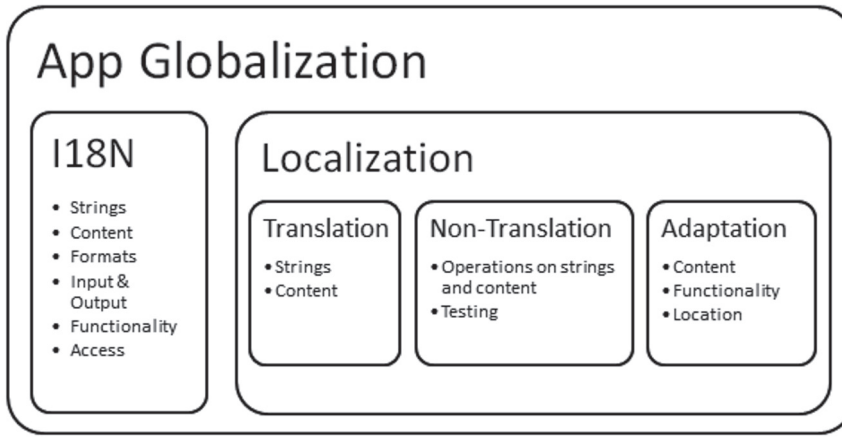

Figure 1.2 Conceptual framework for app globalization

resulting in 'trans-creation'. This controversial term is defined by Torresi (2010: 5) as 'the rebuilding [of an] entire promotional text so that it sounds and reads both natural and creative in the target language and culture'. Some will argue that this definition applies to a step of a standard translation activity, but few will dispute that it is almost never the case in practice as far as the translation of software strings and user assistance content is concerned.

Additional adaptation activities may be related to the actual functionality of an application (e.g. locale-specific resources for a spell-checker may have to be found) or its location (in the case of a Web application). It should be noted, however, that some of these activities may sometimes be omitted from the localization process due to budgetary reasons. While replacing source strings with translated strings is a process that is fairly established and somewhat predictable from a resourcing perspective, adapting functionality and location requires a completely different set of skills and resources.

While the categorization presented in Figure 1.2 may be debatable due to the possible overlap between two categories (e.g. is finding relevant locale-specific details such as a technical support contact email address a translation task or an adaptation task?), it seems adequate to present a wide range of localization activities. While localization is not about creating a full, new product from scratch, it is sometimes necessary to create some parts from scratch to supplement or replace existing translated sections, in order to meet local expectations or comply with local laws and regulations (e.g. when producing an end-user license agreement).

1.4 Who is this book for?

This book is mostly relevant to teachers and students of written translation or multilingual computing courses, newly graduated translators, and even

experienced freelancers. In the 2000s, Esselink (2003b: 7) predicted that 'translators [would] be able and expected to increasingly focus on their linguistic tasks in localization'. Since this book focuses on some technical aspects of localization (over which freelance translators typically have little interaction or control), one may initially wonder how translators will operationalize this knowledge. Three main reasons can be identified.

First of all, the technology that is used during the translation process can be quite complex and therefore difficult to master for non-technical persons. For instance, MT and its myriad implementation details require an ability to learn a new tool extremely quickly without necessarily impacting productivity and quality. Translators who are comfortable updating MT systems with well-defined linguistic assets can therefore add value to an MT-powered localization process (which may or may not require post-editing).

Second, it is now expected that large volumes of content will be translated and checked in limited timeframes, sometimes by having reviewers focus on the most relevant part of this content. For instance, it may be more critical to check the accuracy of warnings sections in installation guides rather than sections describing applications' use cases. Time pressures in translation delivery have been identified as an important factor by Dunne (2011a: 120), who argues that 'in the current market for translation and localization services, time is arguably the most critical constraint'. The increasing use of (semi-)automated data-driven approaches during the translation quality assurance process suggests that both technical and linguistic skills are required to identify those content sections that are worth spending time on. Manually reading a document from beginning to end is no longer practical so new strategies are called for.

Third, the technical complexity of the translation process in localization projects is exacerbated by the amount of people involved. This is becoming particularly apparent when people with no formal translation background or expertise are involved (e.g. crowdsourcing). If translation consistency is a requirement, challenges can be expected when harmonizing terminology or style. Again, effective strategies to quickly check and edit large amounts of content are desirable.

A recent conversation on a public LinkedIn group, however, suggests that it is difficult for some translators to find good training on technical topics related to translation.[16] This situation may be worsened by the fact that translators rarely receive source files to work on. It may be true that in distributed, outsourced workflows (where multiple intermediaries exist between the client and the translators) translators do not generally receive software resource files or even user assistance source files. Instead, they receive project files containing translatable text. However, it may be premature to conclude that the era of versatile, technically savvy translators has passed, especially when translators work directly with clients or when the content to be translated is of a technical nature.

This book has therefore two types of audience in mind: professionals and volunteers. It is written with accessibility in mind so it can be used as a resource

for newly graduated translators who have not received specific training on the localization of software applications and who wish to specialize in this field when starting their professional career. It will also be useful for freelance translators specializing in other fields, and who wish to start translating digital content (such as software products or Web sites). Finally, other professionals working in the field of digital content management (such as technical communicators, app developers or program managers) might also benefit from reading this book. While these professionals would not be responsible for translating the content they produce or manage, they would benefit from being aware of the challenges that have to be resolved downstream. The other audience this work concerns are translation volunteers, specifically technically or linguistically savvy individuals who are involved in non-profit work that requires both internationalization and localization activities (e.g. NGOs, open-source projects). The examples chosen in this book actually have a very strong bias towards open-source technology as a way to give back to the overall open-source community.

1.5 Book structure

The rest of the book is divided into five main chapters and a conclusion. As is customary in this book series, all main chapters include tasks so that readers can actively practice what they have learnt using hands-on exercises.

Chapter 2 focuses on professional practice in order to give readers an overview of some of the technical skills required to be well-equipped when working in the software localization industry. To become a good domain-specialist translator, knowing at least two natural languages and being able to translate well between them is not sufficient. Chapter 2 addresses this gap by introducing basic programming concepts, including text processing ones, so that translators become more comfortable working with fragments of the programming code that is used to write apps. Chapter 2 provides a brief description of software development concepts, programming languages, encodings, strings, files and regular expressions. In order to illustrate some of these concepts with examples, the Python programming language is used. Python is a popular language, which is often described as being easy-to-use, especially for non-programmers.[17] An important characteristic of the Python programming language, however, is that it is currently going through significant changes. Like other programming languages (and natural languages to some extent), the language has evolved over the last number of years to take into account new user requirements. Such requirements have introduced some compatibility issues that are preventing certain users from upgrading to the latest version of the Python programming language. This means that two versions of the language have to co-exist for the foreseeable future. In this book, I decided to focus on version 2.x (where *x* corresponds to a minor version number such as 6 or 7) instead of version 3.x. This choice is mainly motivated by the fact that several *libraries* or *frameworks* (which are collections of existing code functionality) only work with version 2.x. Even though the *older* version is used in this book, it does not mean that the topics covered in this book

will be obsolete any time soon – support for version 2.x has actually recently been extended to 2020.[18]

Chapter 3 focuses on the internationalization issues and solutions listed under *I18N* in Figure 1.2. According to the terminology used earlier (Pym 2004: 9i), the localization process attempts to transform *excluded* locales into *participative* locales rather than *observational* locales. Since this process might involve the delivery of content in multiple languages, content owners must plan ahead to ensure they can quickly cater for all their multilingual customers. This challenge, which is commonly associated with internationalized design principles, is addressed from three different angles in Chapter 3. The first part of this chapter introduces concepts that are related to the creation of a global application, by using a concrete Web application example. Section 3.2 then presents the challenges involved during translation and quality assurance when the software content itself is not internationalized (e.g. problems with text clippings, string concatenation, etc.). Finally, section 3.3 examines how other content types (e.g. user assistance content) can be internationalized as well to ease the translation process (from a time and cost perspective). User assistance is used throughout this book to refer to the informative, textual content that companies or developers produce to document their products or services (including release notes, user guides, tutorials, FAQs and technical support documentation). Poor source quality (e.g. ungrammatical or ambiguous sentences) generates queries during the translation process and culturally-specific content may be equally difficult to translate (e.g. casual style, irony). Some strategies are therefore sometimes put in place to ensure that user assistance conforms with terminological or stylistic guidelines (Kohl 2008).

Chapter 4 introduces basic localization processes, focusing both on translation and non-translation activities that are related to textual content (as defined in Figure 1.2). Some of the translation challenges associated with various content types are highlighted. For instance, specific usability issues arise when working on mobile platforms: should abbreviations be used when translating software strings? Software strings, which are covered first, fall into the category of presentation content that developers produce to make their apps usable. The second part of this chapter focuses on the translation of user assistance content, with an extensive discussion on the role of translation guidelines and automation.

Chapter 5 provides a break between the two localization-oriented chapters, Chapter 4 (localization basics) and Chapter 6 (advanced localization). In this chapter, the main discussion relates to the translation technology that is used to support some of the activities introduced in Chapter 4, including translation management systems, translation environments and terminology tools. An important discussion on machine translation then follows, presenting the differences between MT resource building (whereby an MT engine is optimized for a subsequent, possibly indirect translation process) and MT post-editing (as a direct translation process). Finally, strategies and standards for translation quality assurance within localization are discussed.

Chapter 6 takes up where Chapter 4 left off, by covering the third category of localization activities, which were presented in Figure 1.2 under *adaptation*. In the first section of this chapter, the adaptation of non-textual elements, such as graphics and videos, is briefly discussed. The second section of this chapter provides an overview of various textual transformations, involving advanced localization or adaptation processes. This section contains a brief account of how personalization is slightly changing the way localization is being conducted. The third section focuses on the challenges encountered when the actual functionality of an application has to be adapted to work in a consistent manner across various languages (e.g. by adapting natural language resources such as lists of stopwords required by applications such as grammar checkers or voice commands). Some of these locale-specific challenges supplement the engineering-related requirements listed by Giammarresi (2011: 40) (e.g. import/export methods, text wrapping, searching, etc.). The final section of this chapter focuses on the adaptation of an application's location in order to address issues related to user experience and local regulations.

1.6 What this book does not cover

Many activities can fall within the scope of globalization when a software application or service is made available in new markets. For instance, setting up a local support team (to help users) or a finance team (to process revenue) can be described as global operations. In this book, however, the terms *globalization* and *localization* are strictly restricted to activities that are directly connected to the application and its digital ecosystem, excluding any discussion of hardware-related issues.

This book focuses on some aspects of digital content internationalization and localization, with a strong emphasis on textual content. Both topics are covered from a content processing perspective rather than a project management perspective, since the latter is covered extensively in Dunne and Dunne (2011). Due to space constraints, all content types (especially proprietary multi-modal formats such as Adobe Flash) cannot be covered. Also, it is not clear whether such proprietary technology (which is currently in use to publish videos) will still be necessary to publish or consume multi-modal content in the future, especially with the advent of open technologies such as HTML5.[19] Also, all textual genres cannot be covered in this book. Specifically, video games will not be discussed, since this genre is covered in Chandler et al. (2011). The translation of marketing content is discussed in detail in Torresi (2010) so it will be only briefly mentioned in Section 6.2. Due to the fragmented nature of the localization industry as well as the number of technologies that are being serviced by this industry, it is worth highlighting that this book alone is not key to professional success. It is the author's hope, however, that some of the concepts introduced in this book can be applied and used effectively in some of the situations aforementioned, that have not been covered in detail.

1.7 Conventions

Several conventions have been used throughout this book. Unusual words, terms or examples are clearly marked with the use of *italics*. Characters or phrases that have a specific meaning in a programming context (say, in Python code) are identified with the use of **`bold`**. Links to various resources (such as tools or specific articles) are provided in endnotes. Due to the large number of links provided, the last access date (15 July 2014) applies to all links. A basic Web site was also set up to act as a companion to this book (e.g. to provide a list of errata and links to various resources, including the code snippets used in this book, so that unnecessary typing is avoided).[20, 21]

Notes

1 http://www.gnu.org/software/gettext/manual/gettext.html#Concepts
2 http://www.commonsenseadvisory.com/AbstractView.aspx?ArticleID=1416
3 http://www.cipherion.com/en/news/243-more-irish-hotels-catering-for-non-english-speaking-tourists
4 http://www.libreoffice.org/community/localization/
5 http://bit.ly/x3NmJH
6 http://www.culturalpolicies.net/web/ireland.php?aid=519
7 http://www.culturalpolicies.net/web/germany.php?aid=518
8 http://1.usa.gov/1wzTgsX
9 http://www.oscca.gov.cn/index.htm
10 http://nerds.airbnb.com/launching-airbnb-jp/
11 http://www.telegraph.co.uk/technology/apple/9039008/Apple-iPad-outselling-HP-PCs.html
12 http://www.gartner.com/newsroom/id/2623415
13 http://translate.twttr.com/welcome
14 http://support.microsoft.com/
15 http://www.wordfast.net/
16 http://www.linkedin.com/groups/Why-is-so-difficult-find-44105.S.42456766
17 http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html
18 http://hg.python.org/peps/rev/76d43e52d978
19 A recent announcement by Adobe in fact confirmed it is stopping the development of its Flash Player plug-in for mobile devices, since the alternative HTML5 technology is universally supported: http://blogs.adobe.com/conversations/2011/11/flash-focus.html
20 The code (including commands) is provided 'as is', without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and non-infringement. In no event shall the authors or copyright holders be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with the code or the use or other dealings in the code.
21 http://localizingapps.com