Perspectives on Localization

edited by

Keiran J. Dunne

ata

American Translators Association
Scholarly Monograph *Series* XIII

John Benjamins Publishing Company

PERSPECTIVES ON LOCALIZATION

The American Translators Association Scholarly Monograph *Series* is published periodically by John Benjamins Publishing Company. Since contributions are solicited by the Editors, prospective contributors are urged to query the Managing Editor or Theme Editor before submission. The theme and editor for volume XIII are *Localization*, Keiran Dunne.

Back volumes of the ATA *Series* may be ordered from John Benjamins Publishing Company Amsterdam (P.O. Box 36224, 1020 ME Amsterdam, The Netherlands) or Philadelphia (P.O. Box 27519, Philadelphia PA 19118-0519, USA). Volume I (*Translation Excellence*, edited by Marilyn Gaddis Rose), Volume III (*Translation and Interpreter Training and Foreign Language Pedagogy*, edited by Peter W. Krawutschke) and Volume IV (*Interpreting-Yesterday, Today and Tomorrow*, guest editors: David and Margareta Bowen) are out of print. The following volumes are available:

Volume II *Technology as Translation Strategy*. Guest editor: Muriel Vasconcelles, Washington, D.C.
Volume V *Translation: Theory and Practice. Tension and Interdependence.* Guest editor: Mildred L. Larson, Summer Institute of Linguistics (Dallas, Texas).
Volume VI *Scientific and Technical Translation*. Guest editors: Sue Ellen and Leland D. Wright, Jr., Kent State University.
Volume VII *Professional Issues for Translators and Interpreters*. Guest editor: Deanna L. Hammond, Washington D.C.
Volume VIII *Translation and the Law.* Guest editor: Marshall Morris, Puerto Rico, Rio Piedras.
Volume IX *The Changing Scene in World Languages. Issues and challenges.* Guest editor: Marian B. Labrum, Brigham Young University, Utah.
Volume X *Translation and Medicine.* Guest editor: Henry Fischbach.
Volume XI *Translating Into Success.* Guest editor: Robert C. Sprung, Harvard Translations, Inc., Boston.
Volume XII *Beyond the Ivory Tower. Rethinking translation pedagogy*. Edited by Brian James Baer and Geoffrey S. Koby, Kent State University.

Perspectives on Localization

American Translators Association
Scholarly Monograph *Series*

Volume XIII

edited by

Keiran J. Dunne
Kent State University

JOHN BENJAMINS PUBLISHING COMPANY
AMSTERDAM/PHILADELPHIA

The paper used in this publication meets the minimum requirements of American National Standard for Information Sciences — Permanence of Paper for Printed Library Materials, ANSI Z39.48–1984.

Library of Congress Cataloging Serial Number 87-658269

 ISSN 0890-4111

ISBN 90 272 3189 3 (Eur.) / 90 272 3189 3 (USA) (Hb.; Alk. paper)

John Benjamins Publishing Company • P.O. Box 36224 • 1020 ME Amsterdam • The Netherlands
John Benjamins North America • P.O. Box 27519 • Philadelphia, PA 19118-0519 • USA

Table of contents

A Copernican revolution

Keiran J. Dunne

Twenty years ago, "localization" as a profession and industry did not exist. In the intervening two decades, localization has emerged as a profession related to, but distinct from, translation, and as an industry in its own right that "has grown to the point where the 20 largest IT companies alone are leveraging around USD 1.5 billion a year to generate sales of some USD 15 billion, an incredible ROI of over 1000%" (LISA 2003: 18). Today the total size of the localization industry worldwide is estimated to be approximately USD 8.8 billion per annum (Beninatto and DePalma 2005), and yet despite the industry's phenomenal growth over the past 15 years, localization remains a little-known and poorly understood phenomenon outside of the relatively closed circle of its clients and practitioners.[1] Even among the various stakeholders — clients, vendors, subcontractors, executives, developers, sales and marketing personnel, and project managers, to name a few — there exists no consensus as to what precisely constitutes localization, due in large part to the ways in which it is perceived.

Definitions of localization tend to be contextually bound, reflecting the perspectives of those who formulate them (See Folaron 196–197 in this volume). Those working on the front lines, where translation generally comprises the bulk of day-to-day localization work, may consider localization essentially as "translation on the computer for the computer," to borrow a phrase from industry pioneer Jaap van der Meer (1995: 14). Business executives tend to consider localization through the quantitative prism of return on investment: "There is usually no advantage for a developer in localizing a consumer CD-ROM title or business application for any market unless it can make money in that Market [sic]" (Wilson 1997). Localization project managers, who strive to reconcile the conflicting (and sometimes contradictory) demands of all stakeholders, may view localization as an exercise in herding cats. Consultants and industry experts may take a broader perspective in which localization is all about leveraging source-language development investments to drive revenue, profits and enhanced market share in international markets by "combining language and technology to produce a product that can cross cultural and language barriers" (Esselink 2003: 4). Like a Rorschach test, such definitions provide greater insight into the perspectives of those who formulate them than they do into the phenomenon of localization itself. The very range of perspectives and their contextually-bound nature nonetheless shed light on the structural barriers to a global view of localization.

Indeed, localization does not consist of a discrete process or a defined set of tasks, but rather represents a focal point in the corporate matrix[2] at which various business units, objectives, and processes intersect:

- Development and authoring, which tend to focus largely — and often, exclusively — on the functional characteristics of products to the detriment of linguistic and cultural considerations that may be critical to successful localization efforts
- Sales, running the gamut from international rollouts of clearly defined and planned corporate strategies to ad hoc opportunism
- Marketing, in which messages, materials, and sometimes products themselves may need to be redesigned based on user preferences or cultural norms of the target locale(s)
- Corporate legal counsel, given the fact that localization of a product and its accompanying documentation may be subject to regional and/or national laws and regulations
- Management, which tends to see localization as back-office spending rather than as a wise investment, and thus tries to minimize localization-related outlays at all costs.

Fragmented perspectives on localization are confirmed and reinforced by educational curricula in which foreign languages and translation, computer science, graphic design, as well as business and management tend to be mutually exclusive areas of study:

- Authors and developers often lack expertise and knowledge of foreign languages and cultures (Best 2004) — even of their own — and consequently may fail to grasp the financial ramifications of culturally-bound decisions and practices.
- Executives as well as sales and marketing personnel are often unaware of the ways in which linguistic, legal, and cultural issues can impact a product's usability, image, and even legality, and thus may make commitments to deliver features or functions without a global understanding of what will be required to effectively carry out their directives.
- Translators possess linguistic and cross-cultural expertise, but often lack technical knowledge and an understanding of the market forces and business imperatives driving the decision to localize in the first place.

- Foreign language and translation educators tend to be the products of humanistic programs that focus on literature and literary analysis, and as such often do not possess in-depth knowledge of localization tools or processes, nor of the market realities driving the industry today.

Localization functions as a Rorschach test not only within the context of a given project or company, but also at the level of industries and organizations. In-house clients, outsourcing clients,[3] vendors, individual subcontractors, industry associations, standards organizations, industry experts and educators all bring very different perspectives to bear on localization. The problem of fragmented perspective is compounded by the relative lack of communication between such groups. In sum, localization simply does not lend itself well to being perceived *globally*.

This is perhaps not surprising given the youthfulness of the field and the speed with which it has developed, thanks to the rapid spread of the computer and the advent of the Internet. Fifteen years ago, localization initiatives were largely confined to the realm of IT and large Silicon Valley companies, and localization itself was generally equated with the adaptation of software for sale in international markets (primarily Western Europe and Asia). Until approximately the mid-1990s, "localization" generally meant the translation of software user interfaces and Help, along with any necessary re-engineering that might be required to ensure the correct display of all on-screen information and to maintain full functionality in target-language versions. However, the advent of the Web browser in the mid-1990s, which fueled both the popular explosion and commercial development of the Web, caused a major shift in the scope of localization *across the entire corporate food chain*. Localization is no longer the exclusive province of major conglomerates with brick-and-mortar sales and distribution facilities worldwide. On the contrary, the emergence of the World Wide Web, a virtual "Eighth Continent" inhabited by over a billion online consumers worldwide (DePalma 2002), has leveled the global economic playing field as never before and opened international markets to any company that publishes a Web site — including sole proprietors. Today, any company that conducts business on the Web is confronted by potential localization issues and ignores them at its peril. The spectacular explosion of e-commerce, especially B2B e-commerce,[4] has in turn fueled the proliferation of an ever-expanding variety of "content"[5] to be localized in an ever-increasing number of formats for an increasingly diverse set of users and locales. The rapid pace at which the tools, technologies and scope of localization are all evolving only serves to further underscore the difficulty of maintaining a global perspective on localization.

So what exactly *is* localization? Drawing on definitions formulated by the Localization Industry Standards Organization and experts such as Pierre Cadieux and Bert Esselink, our working definition of localization is as follows: The processes by which digital content and products developed in one locale (defined in terms of geographical area, language and culture) are adapted for sale and use in another locale. Localization involves: (a) translation of textual content into the language and textual conventions of the target locale; and (b) adaptation of non-textual content (from colors, icons and bitmaps, to packaging, form factors, etc.) as well as input, output and delivery mechanisms to take into account the cultural, technical and regulatory requirements of that locale. In sum, localization is not so much about specific *tasks* as much as it is about the *processes* by which products are adapted.[6]

Moreover, localization is but one of a number of interdependent processes and cannot be fully (or correctly) understood without being contextualized in reference to them. These processes are referred to collectively by the acronym GILT (Globalization, Internationalization, Localization, Translation). However, it might be more logical to reverse the acronym: "TLIG" would more accurately reflect the historical evolution of the industry and the sequential way in which practitioners and corporate strategists have become aware of the relative importance of these processes:

- **Translation.** "The world's second-oldest profession" has existed for thousands of years.
- **Localization.** The term emerged in the late 1980s and early 1990s, as software companies first began to attempt the "translation" of their products, and "this work began to be recognized as an industry related to, but different from and more involved than, translation" (Lieu 1997).
- **Internationalization.** The advent of localization in the late 1980s and early 1990s in turn spurred the development of internationalization. "When multiple localization efforts were performed on the same product, it became obvious that certain steps could be performed in advance to make localization easier: separating translatable text strings from the executable code, for example. This was referred to as *internationalization* or *localization-enablement*" (Cadieux and Esselink 2002).
- **Globalization.** In the context of localization, this term refers to the process of conducting business globally, i.e., of selling in international markets. Globalization entails an enterprise-wide focus and set of strategies. On a larger level, globalization refers to international economic and technological integration and the concomitant "flattening" of the world that is driving localization (Friedman 2005).

Figure 1a frames the relationship of the GILT acronym components in terms of their relative dependencies, with globalization as the primary macroeconomic

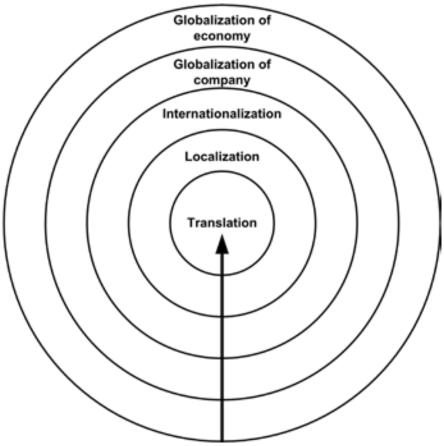

Figure 1a. The interdependence of GILT processes.

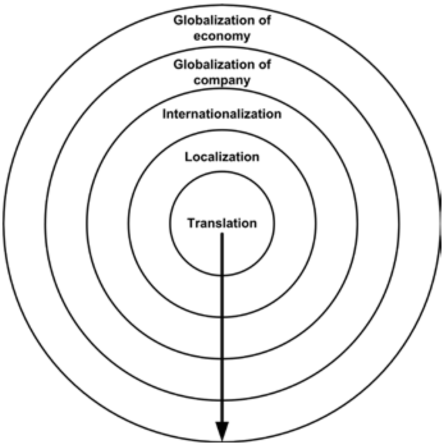

Figure 1b. Path to awareness of GILT dependencies.

and upstream process driver. However, chronological awareness of these processes and their relative importance has historically been gained *from the inside out* (Figure 1b). Early attempts to "transform" programs (and their accompanying documentation) from one language version to another soon led to the realization that there was more to it than mere translation; thus was localization born. Subsequent localization efforts revealed upstream measures that could be taken to facilitate the process, and thus was internationalization born. The need to rationalize the entire development life cycle to support the simultaneous shipment of numerous target-language versions of products in turn led larger software publishers to adopt global strategies designed to facilitate all aspects of the entire process. In this way, globalization strategies are supplanting internationalization strategies at the enterprise level, a process that has been confirmed and reinforced by the disproportionately large profits derived from localization efforts (see DePalma 19–20 in this volume).[7] Driving the progressive strategic shift of planning and effort upstream from translation and localization to internationalization is the larger process of international economic, technological and financial integration, coupled with falling trade barriers, known as globalization.

Over the past two decades, awareness of GILT processes and their relative interdependencies has thus developed progressively, as sequential layers of process have been peeled away and explored like the successive figures of a matryushka doll. However, unlike in a matryushka, in which each subsequent layer is smaller and more narrowly circumscribed than the last, the successive layers of GILT peeled away over the past two decades encompass progressively *larger* scopes (see Figure 2).

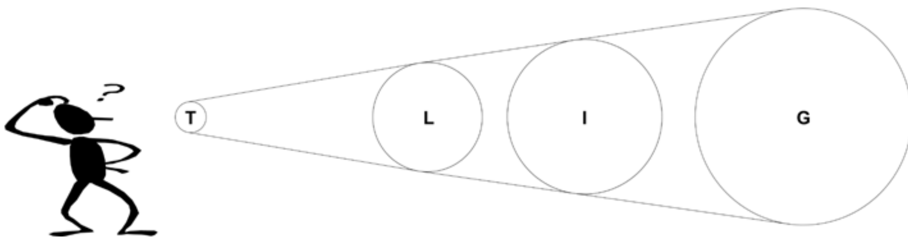

Figure 2. GILT dependencies as they tend to be progressively uncovered.

For the reasons described above, approaches to localization have historically been largely *descriptive* and characterized by incremental improvements, as companies have successively grappled with and resolved issues of translation, then localization, on to internationalization and finally globalization. However, in theory, the effective management of GILT processes requires a *prescriptive* approach, beginning with globalization and progressing downstream through the sequential processes of internationalization to localization and finally to translation. This fact has powerful ramifications for practice, process, workflow and even enterprise structure. As shown in Figures 1a and 1b, there is a chain of dependencies from globalization, to internationalization on through to localization and translation. Working upstream against the flow of these dependencies imposes a point of diminishing returns at each step of the ladder. The relative success of translation and localization efforts depends to a great extent on the successful implementation of internationalization strategies. However, these in turn depend on an enterprise-level commitment to globalization strategies.

It is for precisely this reason that global perspective is so critical in localization. The failure to adopt effective enterprise-wide globalization strategies and to implement appropriate internationalization strategies at the level of product design means that localization processes will be condemned to remain relatively descriptive and reactive, and gains in productivity and cost containment will be incremental at best. Conversely, the prescriptive implementation of globalization and internationalization strategies, given their proactive focus on avoiding problems, will facilitate localization and enable exponential savings by eliminating costly problems *before localization even begins*. To the extent that different participants lack global perspective, they risk being unaware of the downstream impact that their decisions may have on other participants in the process and on the project as a whole. Likewise, when localization is driven by opportunistic sales, as opposed to defined international sales and marketing strategies, and when it is carried out after product development is complete instead of being integrated into the development cycle as a key component of corporate globalization strategies, localization professionals are forced to retroactively fix problems and address the consequences of defective planning after the fact. The bottom line is that in the absence of global perspective, localization is invariably more difficult and more costly than need be the case.

This volume thus focuses on the need for a drastic change in perspectives on localization. The contributors are seasoned practitioners and leading scholars who offer fresh insights and actionable solutions to some of the fundamental issues facing the industry today.

The volume is divided into seven sections. The articles in the first section address the localization business case. Donald DePalma discusses the return on investment (ROI) of localization, based on focused research into the localization practices of 50 U.S. companies. Arguing that localization must adopt the metrics used by the larger business community, DePalma offers a step-by-step process for calculating the ROI of localization in terms of various standard business paybacks, and concludes that when all is said and done, localization is a great bargain. Clove Lynch takes a slightly different tack, exploring the return on globalization management system (GMS) technology investments in support of localization efforts through three Excellence in European eContent Localisation (EEEL) case studies. Lynch suggests that given the critical importance of metrics to building solid business cases on one hand and the relative scarcity of tangible data on the other hand, best-practice models must focus greater attention on the identification, generation and collection of pertinent data that focus on the core GMS value proposition, namely time, cost and quality. Finally, Carla DiFranco discusses the challenges of controlling localization costs. Asserting that most of the fundamental cost drivers are upstream (client-side) issues, she provides concrete strategies for limiting what one might call “cost creep” in localization projects.

The articles in the second section grapple with the slippery notion of localization quality. Scott Bass discusses quality in projects performed on behalf of the middle market — small to mid-size manufacturing companies whose international efforts are more often the product of opportunistic sales than of global sales and marketing strategies. After identifying common obstacles to quality in outsourced localization projects, Bass analyzes and contrasts the quality requirements of clients, agencies and translators, and concludes by proposing a number of practical paths to quality. Keiran Dunne addresses similar concerns in his discussion of localization quality management. He notes that the inherent difficulty of defining what constitutes quality in the localization industry raises questions as to how and to what extent quality can be managed in localization projects. Using the ISO 9001:2000 standard as a frame of reference, he discusses localization quality management in terms of theory and practice, and explores issues that can arise due to the fundamental tensions between customer-focused ISO 9001 principles and the outsourced, sequential localization project model that dominates in the marketplace today.

The articles in the third section focus on the rapidly growing sector of game localization.[8] Frank Dietz' contribution explores the unique challenges associated with game localization that stem from their non-linearity, world-making power, genre conventions and the game development process itself, which often tends to be chaotic. Dietz discusses a number of concrete issues related to interface design as well as cultural and legal considerations, and proposes a number of steps that translators and developers can take to facilitate localization processes. Eric Heimburg addresses the challenges of localizing massively multiplayer online role-playing games (MMORPGs). Because most MMORPGs are open-ended and based on a subscription model, they tend to add new content and activities on a regular basis to retain subscribers. Their sheer volume and constantly evolving nature conspire to make localization more difficult, as do the challenges of enabling players to play the game simultaneously in several languages. However, Heimburg asserts that perhaps the single most daunting challenge of localizing MMORPGs is ensuring high grammatical quality and accurate translations in all language versions. Focusing on Turbine Entertainment's MMORPG Asheron's Call 2, Heimburg discusses the difficulties involved in enabling simultaneous play in English, French, German and Korean and explores the innovative solution devised by the development team, namely a meta-language by which grammatical aspects are embedded in strings in order to enable more accurate translation and a higher degree of grammatical accuracy (even in the English source materials).

The articles in the fourth section address terminology management, which nearly all leading practitioners agree is of critical importance to successful translation and localization efforts, but which few companies have successfully implemented at the enterprise level. Robin Lombard explores the importance of source-language terminology management and draws upon her experience at Microsoft to provide actionable advice for making the business case and for progressively implementing source-language terminology management. Barbara Inge Karsch notes that refining the terminology management process for large-scale, multi-language projects has emerged as a business imperative for software publishers that simultaneously release multiple language versions of their products. Using the J.D. Edwards terminology management system as a case study, Karsch discusses terminology workflow in localization, describing the various steps of the process as well as their characteristics, and modularizing them for application in a generic setting.

The fifth section ponders localization education, which is a particularly critical issue given the structural imbalance between the strong and rapidly growing need for localization professionals and the available supply. Debbie Folaron's contribution addresses localization training and education in academia. Defining the name, terms, and parameters of the discipline is a critical first step, Folaron argues, but establishing curricula that meet both professional and academic objectives is equally important. She then proceeds to sketch the broad outlines of a localization curriculum that does just that, framing her discussion in terms of specific competencies.

The sixth section addresses localization standards and standards organizations. Arle Lommel argues that the increasing importance of standards in localization is being driven, on one hand, by the increasing abstraction of information and the separation of form from content made possible by digitization, and, on the other hand, by the emergence of information as a business commodity with tangible value. Framing his discussion in terms of the shift from knowledge-centered to information business model, Lommel explores the business value of current industry standards. Sue Ellen Wright provides an encyclopedic overview of standards and standards bodies active in the language industry today. After exploring the differences between industrial and language-industry standards as well as the types and philosophies of different standards, Wright discusses coding standards, standards for quality control and quality assurance, as well as functional standards that enable data interchange and interoperability.

The final section focuses on new approaches to the current localization paradigm of *ex post facto* adaptation. Susan Dray and David Siegel explore localization through the prism of user-centered design (UCD). After analyzing problems inherent in conventional product planning and design practices, they explore alternatives offered by UCD, illustrated by representative projects in which they have participated. Localization and user-centered design have much in common, they argue, and increased collaboration between UCD and localization professionals would benefit not only practitioners in these fields, but also those who arguably are the most important participants in the design, sales and marketing processes: the users. Finally, Gregory Shreve argues that human-populated terminology databases and translation memories in the language industry today represent first-generation applications of translation technology whose effectiveness is constrained by the time needed to compile resources, limitations on the availability of linguists, the extremely narrow range of documents that are generally consulted to solve translation and terminology problems, and the generalized reliance on the sentence as the primary translation unit in professional practice. He asserts that the language industry is not exploiting language reuse to its full potential, and advocates a corpus-based process for enhancing computer-assisted translation and localization.

The issues addressed in this volume represent some of the most fundamental challenges to the industry today, which affect nearly all stakeholders, whether they be clients, vendors, freelancers, educators, industry associations or standards organizations. Moreover, they highlight the interdisciplinary nature of the field, and by extension, the critical importance of an interdisciplinary global perspective to successful localization efforts. It is our hope that this volume will broaden perspectives and foster greater dialogue among all stakeholders for the benefit of all.

Notes

1. A search on Amazon.com reveals only six books published since 1999 whose titles explicitly refer to localization of digital content and products (Esselink 2000; Savourel 2001; Symmonds 2002; O'Hagan and Ashworth 2002; Pym 2004 and Chandler 2004).

2. Localization is not confined exclusively to businesses. In fact, NGOs (such as the World Bank) and the open source community both provide examples of non-commercial localization venues. However, this volume focuses on the corporate realm since it is there that the vast majority of localization work is currently carried out.

3. The prevalence of outsourcing has led some to suggest that localization is a commodity activity. However, commotidization presupposes that localization is not a value-added activity performed by a limited number of firms, but rather is a common and easy-to-replicate service in which repeatability and reproducibility are taken for granted. The current absence of standardization in terms of client-side authoring, design and development renders such assumptions highly problematic.

4. Industry research firm IDC predicts that e-commerce will be a USD 7 trillion market by 2007, with B2B e-commerce accounting for USD 2.2 billion in 2005 alone (Crawford 2005).

5. Content is defined as "any digitized information — that is, text, document, image, video, structured record, script, application code, or metadata — used to convey meaning or exchange value in business interactions or transactions" (DePalma 2003: 6). EMC estimates that the volume of data stored on corporate servers is increasing by more than 50% per year.

6. In the absence of effective standardization of client-side authoring, design, and development, localization cannot possibly be reduced to a cookie-cutter process. On the contrary, given the variety in the nature of projects (from voice-over to traditional string-based compiled software user interfaces to Web services, to games, etc.), and given the scope and complexity of the tools used to both author and localize products, it can be argued that localization is both an art and a science.

7. Robert Holleyman, President and CEO of the Business Software Alliance, notes that "[t]he U.S. software industry derives more than half its revenues from exports" (BSA 2005).

8. Industry analyst firm Informa Telecoms & Media estimates the size of the computer game industry at USD 35.3 billion in 2005 and predicts that it will grow to USD 58.4 billion in 2007 (Carless 2005).

References

BSA (Business Software Alliance). 2005. "Testimony of Robert Hollyman, President and CEO, Business Software Alliance, before the House Committee on Government Reform, May 13, 2005." Committee on Government Reform. http://reform.house.gov/UploadedFiles/BSA%20-%20Holleyman%20Testimony.pdf

Beninatto, R. and DePalma, D.A. 2005. "Ranking of top 20 translation companies." Chelmsford, MA: Common Sense Advisory, Inc. http://www.commonsenseadvisory.com/en/research/top_20.htm

Best, J. 2004. "How eight pixels cost Microsoft millions." *CNET News.com*, August 19. http://news.com.com/How+eight+pixels+cost+Microsoft+millions/2100-1014_3-5316664.html

Cadieux, P. and Esselink, B. 2002. "GILT: Globalization, internationalization, localization, translation." *Globalization Insider* 11 (1.5), March 22. http://www.lisa.org/archive_domain/newsletters/2002/1.5/index.html

Carless, S. 2004. "Lost in translation — Japanese and American gaming's culture clash." (Interview with InterOne Inc's John Ricciardi.) *Gamasutra*, January 21. http://www.gamasutra.com/features/20040121/carless_01.shtml

———. 2005. "Informa predicts $58.4 billion game industry in 2007." *Gamasutra*, 24 Oct. http://www.gamasutra.com/php-bin/news_index.php?story=6942

Chandler, H.M. 2004. *The Game Localization Handbook*. Hingham, MA: Charles River Media.

Crawford, A. 2005. "Managing the Tower of Babel." *Globalization Insider* 14 (5). http://www.lisa.org/globalizationinsider/2005/05/managing_the_to.html

DePalma, D. (2002). *Business without Borders*. New York: Wiley.

———. 2003. "Rage against the content management machine." The Multilingual Digital World, 2003 Localisation Research Centre Conference. http://www.localisation.ie/publications/presentations/2003/Conference/Presentations/DePalma%20LRC.ppt

Esselink, B. 2000. *A Practical Guide to Localization*. Amsterdam/Philadelphia: John Benjamins.

———. 2003. "The evolution of localization." *The Guide to Localization*. Supplement to *MultiLingual Computing & Technology* 14 (5): 4–7.

Friedman, T. 2005. *The World is Flat: A Brief History of the Twenty-First Century*. New York: Farrar, Strauss and Giroux.

O'Hagan, M. and Ashworth, D. 2002. *Translation-Mediated Communication in a Digital World: Facing the Challenges of Globalization and Localization*. Clevedon, UK: Multilingual Matters.

Lieu, T. 1997. "Software localization: The art of turning Japanese." *J@pan Inc*, December. http://www.japaninc.net/computingjapan/magazine/issues/1997/dec97/local.html

LISA (Localization Industry Standards Association). 2003. *The Localization Industry Primer.* 2nd ed. Féchy, Switzerland: SMP Marketing/LISA.

Pym, A. 2004. *The Moving Text: Localization, Translation, and Distribution.* Amsterdam/Philadelphia: John Benjamins.

Savourel, Y. 2001. *XML Internationalization and Localization*. Indianapolis, IN: SAMS.

Symmonds, N. 2002. *Internationalization and Localization using Microsoft .NET.* Berkeley, CA: Apress.

van der Meer, J. 1995. "The fate of the localization industry and a call to action." *The LISA Forum Newsletter* 4 (4), 14–17.

Wilson, A. 1997. "The new age of localization." *Computing Japan*, January. http://www.cjmag.co.jp/magazine/issues/1997/jan97/newage.html

PART 1

The localization business case

Quantifying the return on localization investment

Donald A. DePalma

Introduction

This article discusses the return on investment (ROI) of localization, which we define as the process of adapting software, products, documentation, or Web sites to local market needs. According to our continuing research at Common Sense Advisory, localization efforts cost a small fraction of the international revenue they generate. Nonetheless, most firms in the United States underfund their localization budgets.

In this article we recount focused research into the localization practices of 50 U.S. companies. We introduce a step-by-step process for measuring and ensuring the ROI of localization projects, considering a wide range of standard business paybacks and offer formulas for calculating returns. Practitioners can draw on the experiences of others and on these metrics to demonstrate shareholder value to corporate budgeters mindful of post-Enron accounting scrutiny. We define return on investment as follows:

> ROI is a multi-variable, time-variant calculation that will be made by many participants in a value chain. A comprehensive view of ROI should capture the broader range of economic benefits that companies expect from their international investments — including increases in sales and market share, goodwill from better localized branding, lower support costs due to in-language information, and increased customer service and loyalty.

Localization enables international revenue

Localization advocates and practitioners should internalize three bits of data:

- *Disproportionate profits derive from international markets.* Business originating outside domestic markets comprises 40 percent of assets at large companies, but accounts for 45 percent of corporate profits (Gestrin et al. 2001).[1]

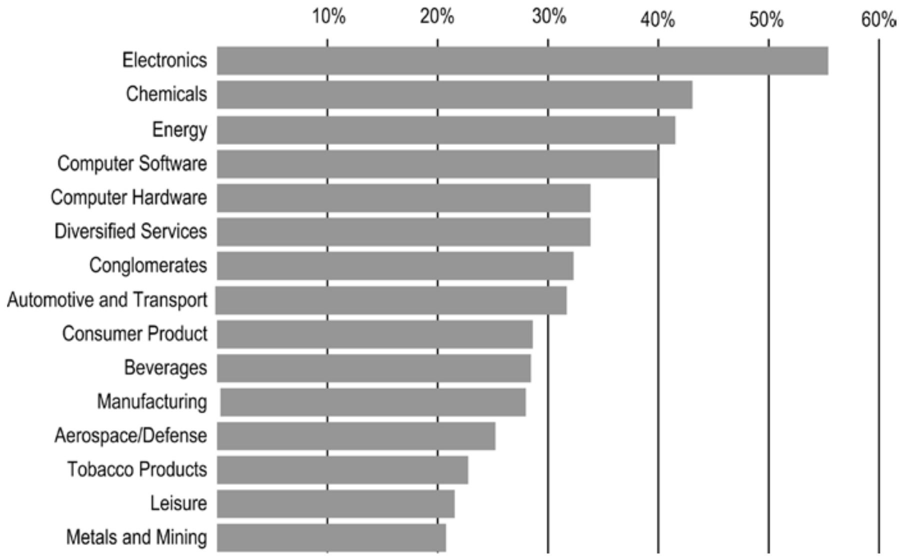

Figure 1. Market segments that earn 20+% of revenue outside the United States.
Source: Common Sense Advisory, Inc.

This favorable ratio should factor into either-or decisions as companies decide where to invest limited corporate resources.

- *Big international payouts require competitive offerings.* Revenue generated outside the country accounts for 20 to more than 50 percent of the total revenue in many market segments in which Fortune 500 American companies compete (see Figure 1). Fortune 100 firms derive on average 28 percent of corporate revenue from non-U.S. business. In each market, these firms do battle with local, regional, and multinational rivals that offer local variants of their goods and services.
- *Localization expenditures exert great leverage.* Our research has shown that small expenditures of 2.5 percent and less of international revenues in research and development, documentation, marketing, and Web development enable companies to adapt products and services for six to ten international markets. This small outlay pales in comparison to the more than US$90 million Procter & Gamble spent merely advertising its Crest Whitestrips in the U.S. market in 2002.

Localization managers seek to raise visibility in the corporate food chain

To understand how companies with ambitions beyond their own markets think about and measure localization efforts, in the fall of 2002 we conducted detailed interviews with 50 managers responsible for setting or implementing localization strategy at U.S.-based organizations (DePalma and Beninatto 2002).[2] This group represented a healthy distribution of vertical markets including automotive, chemical, technology, pharmaceutical, publishing, hospitality, and retail.

Since that time we have periodically plumbed the user community, through both surveys and consulting engagements, to understand whether their situation had changed. We found that it had not. The economic slowdown from 2001 to 2003 caused a hiccup in localization efforts at many companies, thus extending the usefulness of data obtained from this focus group well beyond the one-year window we typically expect. In our discussion of this sample, we note differences between this 2002 sample and subsequent interviews across these markets.

Localization spending has see-sawed through a tough market

In 2002 we had heard enough dismal tales from translation agencies to wonder aloud whether comprehensive localization was an issue — or last year's news. But instead of hearing about huge budget cuts and much diminished localization activity, we were heartened to learn that practitioners accelerated their work to localize products, services, documentation, call centers, and Web sites.

- In the 2002 sample, 57 percent of interviewees said that they were spending more in 2002 than in 2001; another 19 percent told us that their spending held its ground from the previous year.
- By autumn 2003, the pendulum had swung again, with spending plans somewhat down. By mid-year 2004, our conversations with buyers and vendors pointed to another upswing that continued through 2005.

We believe that the 2002 increase was due to postponed investment in localization, while 2003 reflected a period of project review and development. While it is too early to predict its longevity, we think the uptick that began in mid-2004 resulted from an improvement in general business conditions and an increasing realization of the strategic importance of creating a more global product line. In fact, we often hear statements along the following lines.

> "We increased spending 60 percent over last year, mostly due to a push to localize more products. We have to translate more documentation and user interfaces to get deeper into current customers' organizations and to penetrate smaller firms." [Equipment Manufacturer]

> "There's been a dramatic increase in spending over the last year; at least 20 to 25 percent. I can attribute part of the increase to greater global need for localized products and a raised executive awareness." [Consumer Products (Durables)]

At almost all companies with active localization teams, we found that whether spending was up, down, or flat, every firm had a mandate to do more work. Our respondents pointed to many reasons for this change, including increased executive attention to global competition, international marketing, ramping up global customer relationship management, internal reorganizations, and product or company acquisitions.

> "While we're spending less money on localization, we're spending a lot more time on it. We have been expanding internationally at a much-accelerated rate and have found a lot of competition. Our competitors are pushing us toward more localized product." [Food & Beverage Services]

But few companies measure the bottom line for localization

In the unpleasant economic climate of 2001 to 2003, ROI frequently arose as a conference topic, topped the agenda at many executive staff meetings, and inspired reams of articles in general business and trade publications. On the localization front, we found that nearly three-quarters of interviewed companies have not progressed beyond hand-wringing for localization ROI. ROI measurement improved a bit in 2005, and we expect localization efforts to come under the umbrella of regulatory compliance initiatives, business monitoring, and other practices that will support rigorous ROI management.

- *A select few formally quantify return.* Only 26 percent said that they could formally measure and calculate the return on their localization investment. Those undertaking online localization projects were the most evolved, able to measure every interaction that could be monitored by their Web-logging software. The balance, though, simply applied direct multipliers to the initial cost as their way to measure their return: For example, they wanted US$30 back for every dollar they spent localizing.

> "We invested in a Web analytic system that lets us track and measure all the activity on the site by point of entry and point of exit. We can see differences in behaviors, what the overall conversion rates are, and look at the overall business growth. We can measure that change over time and overall gross business change." [Hospitality Company]

> "Our subsidiaries keep half the revenue they earn and send the other half back to corporate. This makes them very motivated to sell a lot of software. Whenever a country unit requests localization, we have our engineers estimate what it will cost. If the country can commit to four times the revenue number, we'll do it. In our more mature markets our return varies all the way from 100-to-1 payback to countries where we are losing money." [Software Publisher]

- *Most don't measure, but feel they have no choice but to localize.* Seventy-four percent of our respondents had a much less concrete sense of what their investment yields, often citing localization as "just a cost of being a global business." Pressured by competitors or the needs of their customers, they know they must support global markets, although they might not be able to perform a cost/benefit analysis for their bosses.

> "We don't know how to measure our localization effectiveness. We don't have anything close to a metric except at the very end of the process; we count the number of support calls. In the long run, ROI is about how much product we sell, but we don't measure that as systematically as we should." [Consumer Products (Durables)]

> "What we choose to localize is ad hoc, and depends on what our channel partners or end users insist on having. In a lot of markets, especially when we enter at the high end, they're satisfied with English. Then they come back and ask for it in their language. Some competitors have taken an everything-in-language approach. While it adds costs, it also adds sales. We've lost deals because we didn't have product in-language and won them because we did." [Equipment Manufacturer]

We also found that few firms measure ROI for the long haul. Just a few respondents ever cycle back to re-evaluate their localization decisions. For the most part, we found that once a company decides to localize, that decision is tantamount to a permanent commitment. When pressed, many acknowledged that it would make sense to review past decisions and some claimed that they have plans to do so.

> "We measure the effectiveness of our in-language training courses and have found that the majority of people experience a 30 to 40 percent improvement on the pre-tests if they are taught in their own language. We see an 80 to 100 percent improvement on the post-test when they're taught in their own language." [Energy Company]

> "If I had my way, we would concentrate on the five markets that really matter to our business, those that account for 80 percent of our business. I would then count on this core to trickle down to other markets in the European Union, knowing that the smaller markets would follow. I don't want to win everywhere; I just want to win where it matters." [Professional Services Company]

All said and done, localization stacks up as a great bargain

Localization does not cost a lot of money. Our interviewees reported that they spent between one-quarter of one percent and 2.5 percent of their non-U.S., non-Anglophone-market revenue per year to localize product documentation, user interfaces, Web sites, and service-related materials for six to ten markets. These numbers correspond to an average outsourced localization budget of a meager US$ 6.5 million. None were able to break out from their overall budgets what they spent for internal localization efforts.

In other words, the measurable cost to enable products or services to be sold in multiple international markets represented on average a mere fraction of the resulting revenue. On average, one-third of that expenditure flows outside the company to localization outsourcers. This number does not include other costs of sale such as local marketing and distribution, but it does demonstrate the effectiveness of localization spending. In fact, one interviewee told us that his company spends more landscaping its campus than it does localizing.

Because the incremental cost for localization is so low, many interviewees told us that they do not separately list localization in the cost of goods. One respondent told us that translation is a line item in many different budgets — and almost impossible to roll up into a single corporate dollar figure without major development work on a variety of financial systems to capture this data.

> "All localization funding is mixed in the manufacturing budget. There's no separate line item for translation that we analyze as a component of a vehicle; we just don't break that out." [Automotive Manufacturer]

> "We don't really track the cost of localization. Our organization is so big and we use so many accounting systems that it would be impossible to consolidate that information." [Electronics Manufacturer]

Furthermore, while our interviewees have a good handle on their external localization costs, very few roll up their localization numbers among applications, business units, or country-units. This failure to consolidate corporate-wide spending hinders insight into the overall localization budget and thus limits negotiating power with external agencies.

Intriguingly, we found that many companies have limited insight into what it actually costs to create the owner's manuals, marketing materials, and Web sites that get localized. They spend enormous amounts on creating this original content, but those costs are buried in the product development process. Localization, on the other hand, is a very visible tail wagging at the end of a development project. Because it is usually accomplished by external agencies that are paid real cash for their work, though, budgets do reflect these costs.

> "I spend a lot of time educating why localization is so expensive. But it's not so expensive compared as to what we spend on any English product. Management doesn't understand the true cost of English. For example, how many pairs of eyes have looked at the English user interface, and how many others have gone over it with a fine-tooth comb?" [Consumer Products (Durables)]

Who's in charge? C-level executives are nowhere to be found

Each conference we've attended over the last several years has hosted a simmering debate among vendors about how to sell to "C-level executives" (that is, CEO, CIO, CFO, and so on). Our interviews demonstrated that this is a moot point because there is little C-level participation in the localization game. Just one interviewee bore the title of vice president or higher. With nary a chief globalization officer (CGO) in the mix, vice presidents typically directed us to responsible staff at the director or manager level. These managers focus their energy on specific deliverables rather than on a coherent or comprehensive localization strategy. Hierarchically, most reported to marketing or product development organizations (see Figure 2).

> "There's not much visibility for my group because it's a cheap service — it's a relatively minor investment so it doesn't get the attention of higher executives. When we talked to one vice president about doing core languages for US$8 million, he looked at international revenue figures and pronounced it a drop in the bucket. I absolutely believe that it doesn't register high enough on the radar because we don't have to spend much to have something to show for it." [Equipment Manufacturer]

This middle-manager spot on the corporate totem pole translates into an issue common to most interviewees: lacking the visibility of a big budget, these localization managers often find themselves selling the need for their operations up,

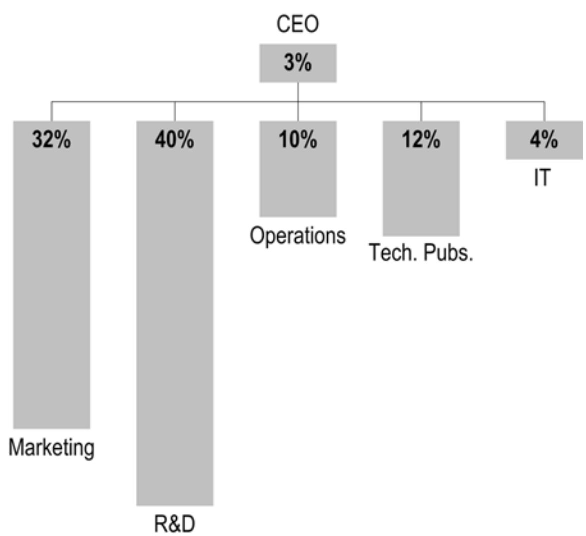

Figure 2. Reporting Structure for Localization Teams.
Source: Common Sense Advisory, Inc.

down, and sideways within their organizations. Ironically, because localization is such a low-visibility operation, some executives see it as an expensive add-on to otherwise well-defined undertakings.

Meanwhile, we have observed the increasing centralization of purchasing control, as a result of which responsibility for buying services is increasingly being assigned to people who don't understand localization issues. This is a natural business trend in which companies will deploy "professional sourcers," a role developed by manufacturers over the last decade to take significant cost out of raw material acquisitions. These experts, armed with a detailed knowledge of their suppliers' economics, have a laser-like focus on extracting the most value from vendors at the lowest price. Their goal is to reduce the total cost of ownership by eliminating fees for sales, project management, quality assurance, and other billable items such as converting formats. They will encourage localization projects to move to standard publishing, technology, and product development formats, thus commoditizing some currently arcane value-added operations i.e., transforming them into easy-to-replicate services differentiated solely by price.

Three obstacles cloud the ROI picture

Many interviewees encountered the same general objections when they "sell" localization in their companies:

1. *"Show me the money before I localize."* The chicken-and-egg problem came up repeatedly in our discussions — "we do not know what benefit there will be from localizing for a market until we do it. And we cannot justify doing it until we know what benefit there will be."

   > "I'm trying to justify to my manager the need for us to continue spending money translating our site into seven non-English languages. Currently, the selection of native language preferences is not very high on our site. Only about 12 percent of our audience has selected to receive information in a language other than English and the qualitative survey we did with our members generally agreed that English is acceptable on the site." [Computer System Manufacturer]

2. *"English works well enough for target audiences."* The more products cost or were used by sophisticated multilingual audiences, the less likely companies were to see the need to localize them. For example, our respondents have found that English works better for some industries — such as science, banking, back office, and software — than it does for consumer packaged goods or home electronics.

> "Our software is aimed at a fairly sophisticated audience, so the higher their sophistication and education, the less need for localization. For example, if someone is mapping the human genome, they're [sic] very comfortable speaking English. It is a different story on the client side, though, where the software has to be localized. The more clients we sell, the more servers we sell, and there's much less need to localize servers." [Software Publisher]

3. *"The U.S. business case works everywhere."* Having spent big-time dollars on focus groups and market research in the States, many firms don't duplicate that outlay to make the case for other markets. While they find non-U.S. revenue desirable, they feel that the business case should derive directly from their U.S. market experience. Until they can accurately and confidently quantify localization benefits through ROI models, they will continue to use the U.S.-focused business case.

> "We don't want to spend much on information about international markets, especially since we know the program worked so well in the States. We'd like to extend a successful e-mail marketing program to 12 international markets, but we really don't think that we should have to spend more than a few thousand dollars total to get detailed information on e-mail and privacy regulations, consumer attitudes, competitive offerings, and a range of other details on those dozen countries. It worked in the States." [Interactive Marketing Services Agency]

Interviewees have mixed emotions about localization suppliers

In 2002, two-thirds of our respondents used external agencies to perform their localization work as a matter of policy. They typically cited one or more reasons, including: (1) Localization falls outside their core competencies; (2) outsourcing lets them better respond to the ebb and flow of projects by bringing in resources only when needed; and (3) it costs less than staffing up internally, an important issue for companies trying to strike the right balance between fixed and variable costs.

By 2004, the outsourcing figure had jumped to 87 percent while discussion centered around even more cost reductions. Planners switched their attention from Dublin — the traditional destination for localization projects — to lower wage markets like India, the Czech Republic, and Argentina. Loyalty to longtime suppliers won't slow down this exodus. While we have found that our respondents were satisfied on average with their localization suppliers, only a few interviewees characterized themselves as being "very satisfied." That's not enough to tie buyers to their suppliers for life.

Our respondents did acknowledge that they bear some responsibility for their outsourcers' performance. They confessed to giving their suppliers little formal direction in the form of style guides, controlled language, or terminology management. Their review process substituted subjective, random checks for formally defined, objective quality audits.

Low automation quotients characterize most respondents' efforts

Because our interviewees find themselves pushed to localize more material for each dollar, we expected that productivity-enhancing tools would be a top priority. They're not.

In our research we have tracked three categories of tools:

- *Translation memory.* In 2002 we found that only 46% of our interviewees employed or required their suppliers to use translation memory from companies like Atril, SDL, and TRADOS. That number has been creeping up, but does not yet qualify as indispensable at most companies. Improved usability and the integration of server-based translation memory with content management systems should improve market acceptance.
- *Machine translation.* In 2002 we found very limited use of computer-generated translation from companies like SDL and Systran. U.S. government investment in machine translation solutions to leverage limited human resources in Arabic and other strategic languages raised the profile in 2003. End-to-end solutions entering the market in 2004 promise to make machine translation a more viable option in narrowly defined content categories.
- *Globalization management systems.* While they were hanging on by their fingernails through 2003, GMS solutions from independents like GlobalSight and Idiom claim to be thriving in 2004, while larger suppliers SDL and TRADOS point to strong sales and a strategic role in large accounts. Nonetheless, buyers have not flocked to these solutions. In 2003 we counted 50 active GMS installations worldwide, largely in high-technology companies. By 2004 we observed penetration into other market segments, including automotive, publishing, and finance.

Conclusions from our interviews

From our discussions with executives and managers of localization initiatives as U.S.-based operations, we conclude that:

- *Localization managers need to demonstrate ROI.* Payback on any kind of investment is on everyone's agenda, but most localization groups have yet to master the business art of quantifying the value of their efforts. Across the board, our interviewees are being asked to do more with less, and to prove the value of their localization projects.

- *Localization providers fail to excite their customers.* While most of our interviewees depend on external resources to do most of the work, they want lower prices and higher quality.
- *Automation lags behind other enterprise functions.* Beyond translation memory, localization technology has yet to gain traction in the marketplace. Many companies fail to provide their outsourcers with basic necessities such as style guides, glossaries, and managed translation memories. Productivity suffers, and costs escalate.

Localization must transfer from liberal arts to business school

Even with such excellent returns, the pressure to do better and more for less money will not go away. The balance of this article focuses on meeting these business demands. We expect that two dynamics will drive practitioners from budget beggar to valued business partners as they:

- *Establish localization value in business terms.* Most companies marginalize localization as a tactic for supporting international marketing, sales, and customer support. This is an unfortunate reality that those responsible for localization must get used to. They will need to show payback in quantifiable terms, which demonstrate that localization makes more money than it costs. Even then, they can expect that executives with itchy delete-key fingers will ask for a cogent year-over-year cost/benefit analysis.
- *Optimize spending.* Even though outlays are minimal, bean counters will scrutinize localization as an expense to be minimized. Accountants will analyze these efforts alongside their examination of procuring raw materials, keeping the physical plant in top running condition, and even keeping restrooms stocked with liquid soap and toilet paper. These inevitable comparisons will force localization managers to cut costs and increase productivity even more, thus driving fundamental changes in buying behaviors and everyday practices.

In this penny-pinching milieu, successful localization managers must start by creating a simple cost-based model to measure spending effectiveness. Over time, they can expect to be tasked with creating ever more sophisticated ROI models that measure a wide range of tangible and intangible benefits. The following sections provide the groundwork for this evolving model, suggesting three critical steps to bringing localization into the business fold (see Table 1).

Table 1. Reality check for localization: Shareholder value must precede all else.

	Champion business metrics for localization	Crunch the numbers year in and year out	Estimate the price tag of not localizing
Focus	Express desired results using core business metrics	Calculate the actual cost of localization and the estimated return	Determine potential losses of not localizing — share, brand, revenue
Status Quo	Decisions are taken on tactical grounds with little attention to ROI	Companies show an irregular, even erratic lack of ROI calculation	Most budget reviews focus merely on the cost to localize
Future	Localization enters business planning, viewed as part of each firm's international business strategy	Calculating localization ROI becomes a business discipline	Nuanced analysis of the risk of not localizing complements "what if?" analysis elsewhere in the business

Source: Common Sense Advisory, Inc.

As companies gauge the payback they can expect from localization, managers will find themselves in a fundamental conflict of interest: the people who have the most to gain are usually the ones asked to perform ROI analysis. That means that the localization team itself is often asked to quantify the potential return on its efforts. In very large organizations software suppliers, service providers, or consultants are asked to conduct the analysis (the irony of this complaint is not lost on the author). The best way to manage this reality is by developing an objective, business-driven case for localization.

These ROI exercises typically measure payback in a two-dimensional form; that is, how much profit or cost savings result from what they spend on a project? What these calculations typically miss is the indirect and non-revenue returns associated with activities like improved customer self-service, branding programs, and regulatory compliance. In our discussion of ROI, we suggest that localization managers adopt this broader measure of the potential returns.

> ROI is a multi-variable, time-variant calculation that will be made by many participants in a value chain. A comprehensive view of ROI — what we call "360° ROI" — will capture the broader range of economic benefits that companies expect from their international investments — including increases in sales and market share, goodwill from better localized branding, lower support costs due to in-language information, and increased customer service and loyalty.

Step 1: Adopt the metrics used by the rest of the business

In today's economic climate, "nice to have" projects don't last very long. That means that managers have to shift attention from the relatively simple task of

Table 2. Sample business goals for localization projects.

	Business Goals	Measurability
Revenue & Share	• Boost profitability of international operations • Increase revenue from existing markets • Grab share from competitors • Own a segment (e.g., mid-range products) • Enlarge client base in existing markets • Protect share in existing markets • Acquire revenue from new markets • Accelerate time to market for new products • Meet regulatory requirements	With few exceptions, planners can easily quantify these business goals, but should establish baselines before starting projects. They need numbers for revenue, market share, and product development from the CFO and colleagues in sales and marketing.
Customer Service	• Boost lifetime value of a customer • Improve customer loyalty • Increase customer retention • Reduce cost through localized self-service • Establish competitive service levels • Support business buyers with localized service • Decrease liability through more effective communication • Lower administrative and transaction costs	With the exception of direct support costs and customer value, attaining these goals requires more effort. Relative goals such as service levels can be compared over time to market rivals. Localization executives should involve colleagues in domestic marketing and customer service in their analysis.
Branding & Intangibles	• Improve brand awareness • Increase brand value regionally or internationally • Manage brand better and more efficiently across markets • Associate brand with local culture • Adapt offerings to local needs	These are the "softest" of business goals. To gauge progress, planners need to invest in surveys and focus groups with customers and prospects. They should work with company's marketing and branding experts to understand brand and mindshare issues.

Source: Common Sense Advisory, Inc.

getting something localized to doing so in the context of maximizing shareholder value. They must determine the ways in which localization spending contributes to and furthers business goals. In this more rigorous business climate, they need to align every localization decision with a measurable element of their company's business strategy (see Table 2).

Step 2: Crunch the numbers scientifically and comprehensively

International business is not a one-time event involving market entry or simply building some stores. Rather, it is about a process that begins with planning,

Table 3. Calculations for determining ROI on localizing for a market.

	Cost basis ROI	Continuing cost basis	Return on equity
Formula	revenue – cost to localize	revenue – (cost to localize + cost to maintain)	revenue – (cost to localize + cost to maintain) – expected loss – (cost of equity * economic capital)
Input to formula	For products, the cost to translate interfaces, documentation, and product information. For marketing, the cost to translate and localize marketing materials and Web interactions.	Builds on Cost Basis ROI, but with the added expense of keeping products, market materials, Web sites, and support current with the evolving product and marketing messages.	Extends the Continuing Cost Basis, but factors in losses while building market momentum and balancing the opportunity cost associated with this project as opposed to a competing effort.
Pro	Provides a useful point-in-time analysis to the impact of localizing a product, service, Web site, or other information	Captures the ongoing cost of maintaining and enhancing a product, service, or online presence in other markets	Characterizes the actual cost to a company of earning revenue for a given project. Return on assets (ROA), net assets (RONA), and equity (ROE) are accepted measures.
Con	Fails to account for the ongoing maintenance required for long-term business commitments	Fails to capture the opportunity cost of supporting one initiative or market over another	Requires more information and analysis than most firms collect or apply to localization programs

Source: Common Sense Advisory, Inc.

continues through product development, extends to deployment and in-market support, and even includes the retirement of an offer.

Keeping this life cycle in mind, localization managers need be encyclopedic in cost estimates, including the cost of equity and opportunity (see Table 3). In the rest of the company, the most common approach is to use cost-basis accounting in which the economic benefit of a localization activity will be the income expected from the activity minus the cost to develop.

Adding maintenance, expected losses, and the cost of capital to the cost-basis calculations will build a business case that should stand up in most boardrooms. Unfortunately, we found that details about the full life cycle of localization and actual results are often not readily available, so managers will have to dig up those numbers.

This quest for hard numbers is a good opportunity for localization teams to interact with fellow employees whose job it is to monitor, capture, and manage these kinds of statistics for other parts of the business. These people work in functional areas such as market and business intelligence, data warehousing, knowledge management, and collaboration. They will need to take advantage of their investment in technology and process improvement to measure the effectiveness of corporate programs that calculate the return on localization efforts.

The bottom line: How much return should there be on localization?

Unfortunately there have been no successful initiatives for localization ROI metrics coming from governmental agencies, regional organizations, or industry groups. However, we can recommend the following best practices for determining how much to spend on localization and what to expect in return:

- *Follow the lead of other groups.* Don't set ROI in a vacuum. Localization owners should work with their CFO and other executives to meet corporate expectations. They should consult with colleagues in marketing, sales, manufacturing, and customer service to understand what they are doing. In the final analysis, the most credible ROI calculations and target returns for localization will be those used throughout for mainstream corporate functions.
- *Set simple, realistic goals.* One interviewee uses a threshold of four-times the cost of localizing a product to prepare it for a market, thus yielding a 30 : 1 return on spending his development funds. While this doesn't factor in the cost of marketing, distribution, sales, and support, it allows the corporate "investor" to triage proposals. He tells market advocates "don't bother asking unless you can show me a 30 : 1 return." This multiplier approach offers a good finger-in-the-wind metric.
- *Compare localization spending with international returns.* While all our interviewees would have preferred having bigger budgets so that they could localize everything, they all seemed to survive — and many excel — with an average budget that equals one-quarter to 2.5 percent of their non-domestic revenue. Many executives will find this ratio of what it takes to enable international revenue easy to understand, even though they will feel obligated to whittle away even at these small numbers.

Will spending ten times as much yield an order-of-magnitude greater return? No. Bigger budgets would enable companies to localize more products for more markets to a deeper level. Sooner or later, though, they will reach the point of diminishing returns. While satisfying German or Japanese customers, inordinate attention to their needs could jeopardize the viability of core products in English-speaking markets where competition might be more fast-paced; delay products past their market-driven "sell-by" date; or limit the introduction of new features.

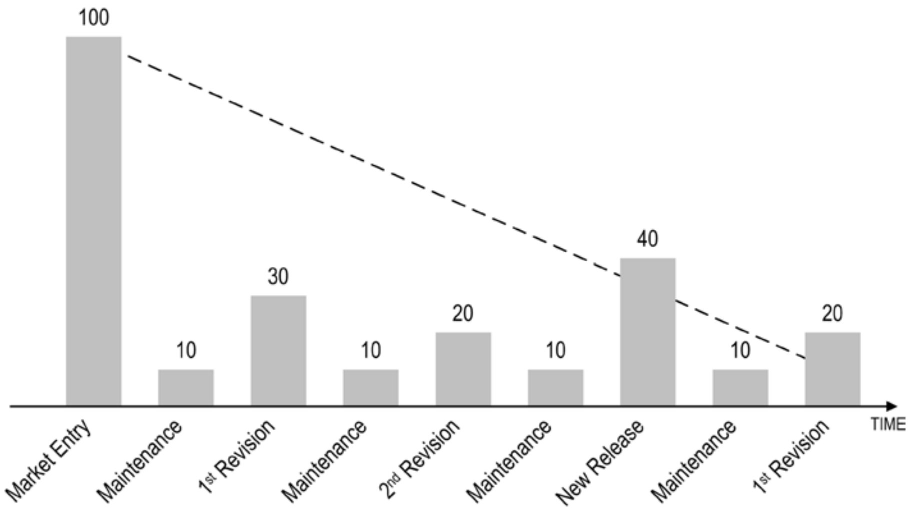

Figure 3. Returns differ based on when you measure them.
Source: Common Sense Advisory, Inc.

Finally, executives need to make sure that their calculations reflect varying levels of investment over the project life cycle (see Figure 3). They will find themselves spending a substantial amount on localization just to enter a market, prepare a product, or localize a Web site. That first outlay will mean that early ROI calculations will be less attractive than later ones, but subsequent reckonings will be more favorable. For example, if productivity increases or external prices fall, ROI goes up. In this figure, 100 represents the initial expenditure to enter the market, while other bars list the relative cost of performing critical functions over the life cycle of a product. For example, yearly maintenance may cost 10 percent of what a company laid out to enter a market.

Richer economic models are needed to better measure and justify localization

Once planners have put together a two-dimensional cost-basis ROI, they should stretch their analysis by moving toward a broader array of incremental business goals, metrics, actions, and targets that demonstrate success (see Table 4). Over time, as they gather more data from monitoring their company's localization efforts, they will begin to be able to plug actual results into standard corporate formulae for measuring return.

Table 4. Sample international retail business goals and metrics to measure them.

Metrics	Actions	Sample target
Increase revenue from international markets	Measure improved sell-through, retention, or customer loyalty. One approach would be to create an A–B split, directing some prospects to an in-language buying experience and others to an English-only venue. This will move the discussion from a vague analyst's proposition to an actionable one.	Double the percentage of people who buy after perusing products
Increase customer loyalty	Business studies show that even a five percent reduction in customer attrition can increase profits by 25 percent or more. Companies increase retention by adapting their offers to those buyers' language and buying motivations, using a variety of systems like personalization, data warehouses, and customer relationship management technologies.	Cut the number of international customer defections by 10 percent
Increase value of brand	Intangibles like brand and goodwill remain the hardest to measure. Gauge the effectiveness of localization spending through studies in unaided brand awareness, advertising recognition, purchase intent, and message association. While it is difficult to associate dollar values with these items, they will at least be stated as recognizable assets that could increase as a result of localization efforts.	Increase unaided brand awareness in local markets by 25 percent

Source: Common Sense Advisory, Inc.

Step 3: Estimate the price tag of not localizing

Larger, higher-GDP markets such as Germany and Japan — where companies will likely find powerful local competitors — have more demanding customers. For these geographies, comprehensive localization will become a basic cost of doing business. Other markets — such as Central Europe and Brazil — will insist less on perfection, finding satisfaction in sincere, albeit sometimes mediocre, efforts. Executives should expect that even these tolerant markets will become less forgiving over time as they move from being prospects to buyers (see Table 5). Their ROI calculations should be sensitive to assess the potential damage of not localizing. They should factor in the decreased competitiveness, lower usability, and higher customer service costs of products not adapted to international market requirements.

Table 5. The cost of localizing — and not.

	Marketing	Sales	Usability	Support
Out-of-pocket expense for localization	Translation and adaptation costs for marketing materials such as printed collateral, Web sites, and advertising	Engineering remediation to adapt transaction and operational systems	Translation and adaptation costs for owner manuals, FAQs, and online assistance	Translation costs for FAQs; in-language search; knowledge base; diagnostics
Costs of not localizing	Local, regional, and more localized international competitors gain share; sudden loss of market share; customers develop loyalties to a competing brand			
	Limited reach of marketing programs	Sales limited to those who are comfortable in the nuances of the English language	Bad reputation for usability; brand damage; and higher support costs	Frustrated customers; expensive problem resolution through call centers

Source: Common Sense Advisory, Inc.

Unlocalized products levy a usage tax on international customers

As companies target international consumers or new buyers deeper in organizations, they will find that these new audiences may not respond to their marketing messages or even be able to use what is on offer unless they have localized products and supporting information to the language, environment, and business practices of those markets. Whether they can — or will try — depends on what Common Sense Advisory calls the "localization tipping point":

> The localization tipping point is the crossover at which adaptation becomes mandatory for a given market. Factors such as the educational level of consumers and business buyers, the required level of regulatory compliance, and the cost of product ownership will drive companies to tailor products, services, marketing, selling systems, documentation, Web sites, and other materials to specific market needs (see Figure 4).

The tipping point is not an absolute or permanent calculation. It varies by nation, by individual product, by consumer, and even by sales channel. For example, buyers may be more linguistically liberal online than they are in the supermarket. This crossover point will also move over time, such that a decision not to localize today may be wrong in six months because consumer attitudes, competition, and business pressures will not stand still. The unrelenting push deeper into retail channels and into transnational supply chains will expose more non-Anglophones — usually unfavorably — to U.S. products that do not meet their needs or expectations.

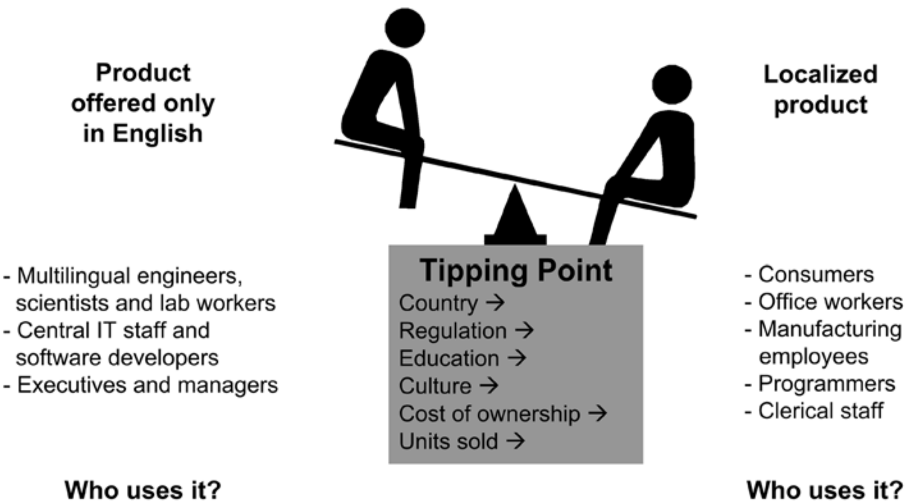

Figure 4. The tipping point will favor localized product.
Source: Common Sense Advisory, Inc.

- *Consumers demand fewer complications in their lives.* The closer a product or service is to a consumer or worker whose job does not require English, the more likely it is that they don't feel comfortable enough to make decisions based on their understanding of that language. For example, if they use a product while wearing a bathrobe, stuck in traffic, or shopping for groceries, they don't want or need the added burden of understanding English.
- *Multinational procurement groups buy for offices around the globe.* While a company selling its products internationally may not have a single office abroad, its buyers often do — where they are subject to local language legislation and workforce realities. Localization thus becomes a requirement. Even if such buyers ultimately opt for U.S.-packaged offering, their decision matrix will include the availability of localized product.
- *Competitors offer localized products.* Wherever buyers can choose among otherwise equivalent products, companies intent on selling globally don't have a choice about localizing. One producer of telecommunications gear told us that his firm won and lost deals on the basis of having or not having local language support. What often tipped the scales was the cost of labor — the salary of a bilingual engineer to install and maintain the system could be two or three times a technician's wages, especially in markets like Mexico and Russia. Products with localized interface and manuals are ultimately cheaper because they let the buyer hire a less expensive person — and thus substantially drive down the cost of owning and operating products.

Experienced corporate buyers will realize that unlocalized products cost them more to install, manage, and provide training for than localized ones. They will translate the lesser usability of these un-adapted products into a higher total cost of ownership (TCO). This usage tax points to an oft-missed reality — ROI is a two-way street. Yes, suppliers expend budget and dedicate other resources to localize their goods or services. However, their customers pay a usability tax to deploy the product, with the rate varying by the level of localization. More localization levies a lower tax, while less localization increases it. This hidden usage tax will show up in the buyer's TCO analysis for buying (or not buying) unlocalized products. Over time, this dynamic will tip more consumers and businesses to favor localized products as they strive to lower the cost of doing business.

Establishing ROI means more work for localization executives

Because most companies we have encountered practice little if any long-term ROI analysis, we expect that many localization professionals will find this approach a sea change in measurement. To justify continued localization, many just said "we've always done it" or "once we make the decision, we're always there." Few companies analyze their market successes or failures, so they can't quantify the value of their investment now. The decision to stay the course is often one of inertia, momentum, or anecdotes, backed up by little if any quantifiable evidence. In an era of massive budget cuts and frequent reorganizations, such complacency is untenable.

Systematic, continuous measurement will help localization managers establish how products, campaigns, channels, documentation, and call center over time contribute to corporate business goals like more revenue, lower cost of sale, or greater brand awareness. Ultimately, the data and the algorithms they develop will enable "sunset budgeting" — that is, the year-by-year determination of the right spending levels, including zero, for each market and every activity in that market. How should companies proceed?

One of the biggest challenges faced by localization executives will be promoting awareness of the importance of what they do. This points to the supreme irony of localization: Practitioners are paid to communicate a value proposition to prospects, customers, partners, and employees; yet they seem incapable of translating their work into what management perceives as valuable and worth every penny spent. Localization teams need to reinvent their persona and describe what they do in business-critical terms.

The immediate task facing localization managers is to understand their firm's business goals, determine where their efforts fit in that context, and "market" themselves accordingly. This communications effort will go a long way toward transforming localization from being perceived as an irritating, costly step in product development to an integral part of every corporate project with paybacks that can be measured across full 360 degrees of business value.

A critical first step is to present their case to the executive team, board, or business decision markets. They need to get on their calendar with a “localization for dummies” presentation that presents their work in the context of corporate product development, global marketing, and customer support. Rather than drone on with tired statistics about the shrinking contribution of the U.S. economy on the Internet, present compelling numbers such as Templeton’s international-asset-to-profit ratios.

1. *Put localization in the context of shareholder value.* Localization executives need to demonstrate how corporate goals such as “own the mid-range market” and “increase the lifetime value of customers” relate to their work. They should use the language and tools of corporate strategists to discuss issues such as how localization increases look-to-buy ratios and limits customer defections. They need to focus attention on which localization efforts have the greatest return for stated business goals, leveraging case studies such as those described in *Business Without Borders: A Strategic Guide to Global Marketing* and others uncovered by Common Sense Advisory and other localization-focused research firms.
2. *Identify known reference points.* Localization budget owners need to compare their spending to that of other parts of the business — R&D, documentation, marketing collateral — so that decision makers and budgeters have a context for appreciating their efforts. To put a fine point on the comparisons, they should dig into operational budgets to uncover how much their company spends on non-revenue related essentials like toilet paper or landscaping services. Whichever budget they choose to compare, this perspective will demonstrate the relatively low outlay that drives such large international returns.
3. *Pitch the bigger picture.* The CEO and Board of Directors probably think that all that is involved in localization is translating documents. To complete their Localization 101 education, localization executives should emphasize the importance of “communicating the company’s messages” to their prospects and customers across the whole value chain. This exercise has the potential to extend their responsibility to non-linguistic efforts such as validating the usability of unlocalized, home-market products outside the United States, getting involved in high-value marketing efforts, and over time becoming a critical business asset.

This exercise will not insulate localization from budget constraints and, paradoxically, this new prominence might create more work. Even more cost-cutting pressures will come to bear on localization projects as companies struggle to regain profitability and re-establish shareholder confidence. Despite the goal of recasting localization as an investment with big payback, the reality is that everyone will always view this as a cost first, then an investment. At the end of the day, localization is a cost. Like all costs it has to be rationalized, decreased, or otherwise optimized.

Notes

1. The authors analyzed the results of 246 of the top 500 multinational enterprises.

2. We collected corroborating data in the fall of 2003 for *Real World Enterprise*, published in January 2004 and throughout 2005 for research on the changing supplier landscape, the buyer reaction, and the changing calculus of ROI.

References

DePalma, D.A. and Beninatto, R.S. 2002. *Beggars at the Globalization Banquet*. Chelmsford, MA: Common Sense Advisory, Inc.

Gestrin, M.V., Knight, R.F. and Rugman, A.M. 2001. *The Templeton Global Performance Index 2001*. Oxford: Templeton College, University of Oxford. http://www.templeton.ox.ac.uk/ubinfo.asp?PubID=1015

GMS technology making the localization business case

Clove Lynch

Introduction

The role of return on investment (ROI) as a determining factor in localization initiatives is well documented (Schäler 2003: 10–11; DePalma 2002: 230–244). However, documentation of the return on technology investments in support of localization initiatives is relatively scarce. The implementation of Globalization Management System (GMS) technology is slowly producing both qualitative and quantitative ROI measurements, though much of the available data is anecdotal or commercial in nature. Fortunately, some published business case studies have emerged, providing answers to questions such as how much a company can expect profitability or market share to increase by investing in GMS or other global content management technology. Three studies are analyzed here to illustrate how or whether they measure the ROI of GMS technology in the context of global content initiatives.

Background

Content varies by industry and application, but can be broadly defined as media with asset value in corporations, institutions and organizations. More specifically, content is defined as "any digitized information — that is, text, document, image, video, structured record, script, application code, or metadata — used to convey meaning or exchange value in business interactions or transactions" (DePalma 2003: 71). Classified at a high level, *content assets* are digital media, non-digital electronic media and non-digital print media. At a lower, functional level, recognizable content assets include reports, manuals, news feeds, faxed communications, emails, Web sites, audio recordings, images, software resource files and many combinations of the same. The value of content assets is a function of the following factors (among others):

- *purpose* — marketing communications (corporate), security bulletins (government), grant proposals (NGO/institutional), etc.;
- *format* — paper, analog audio, digital print, digital audio, digital multimedia;
- *longevity* — perishable (e.g., news feeds), non-perishable (e.g., product documentation, annual reports);
- *re-usability* — alternate use of content assets in contexts other than those for which they were originally intended (Wagner 2002: 5–6).

Content Management Systems (CMS) evolved as a response to the growing need for centralized control over valuable content assets in large organizations. Central storage, access control, business rules, workflow and authoring tools were integrated to create large software applications capable of "enabling the production, maintenance and deployment of content" (Nguyen et al. 2002: 10). Some CMS solutions offer integrated language management functionality by assigning language labels (a.k.a. metadata) to content and by ensuring that the application infrastructure is fully internationalized, i.e., that all core functions (input, output, etc.) can be performed in as many languages as the application claims to support. However, as globalization is often not a core requirement in CMS deployments, CMS applications are not typically designed to offer the depth of localization-specific functionality found in GMS systems. This has led to demand for both integrated GMS-CMS deployments as well as an increasing number of hybrid systems, sometimes referred to as Global Content Management Systems (GCMS). Figure 1 illustrates

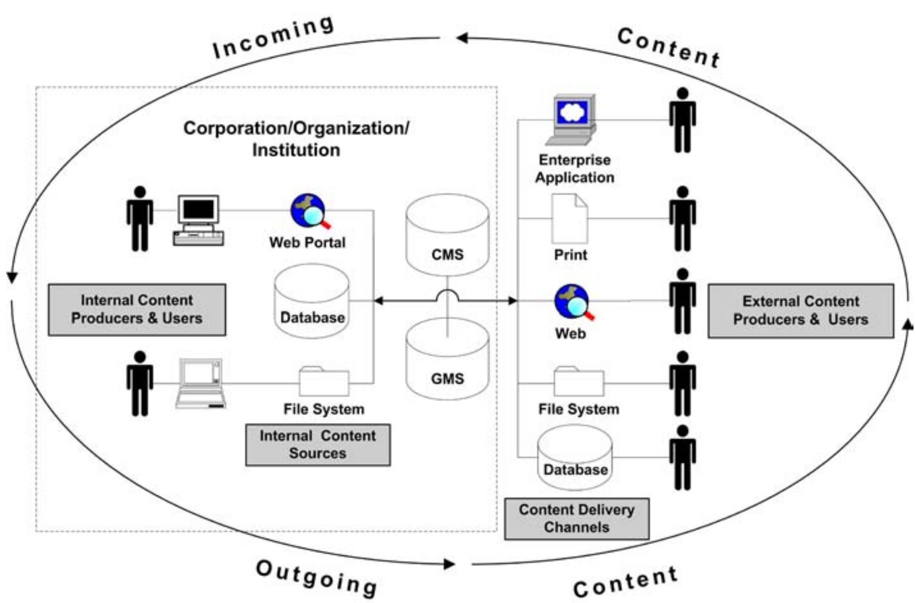

Figure 1. Content management life cycle with CMS-GMS deployment.

a typical integrated CMS-GMS deployment, with a combined system managing a synergistic content flow between business users inside a company or organization who both produce and consume content, and external business users who also produce and consume content.

Localization is an additional investment that content owners make in order to increase the accessibility or commercial viability of high-value content assets, and/or to comply with legal requirements (e.g., in the case of regulated industries such as pharmaceuticals and medical devices). The cost of localized content is typically many times that of the source content, and can be calculated using unit-based rates (e.g., per word, per line, per page, etc.) or simply using hourly rates for processing given units of source content. For example, the cost of content localization could be expressed as:

*(units of source content) * (localization process costs) * (number of target languages)*

Obviously, this is a simplified example for purposes of illustration. In practice, localization costs vary tremendously based on source content type, target language-specific requirements, and a large number of variables such as *content localizability* (whether localization is possible or appropriate), *schedule* (which can drive rates up and impact work patterns) and *volume* (which also impacts rates and processes). An important variable affecting process costs is *change management* resulting from source content updates, which can occur often in localized content life cycles.

The added investment in content localization comes with the expectation of increased return in one form or another. In commercial scenarios, the return is typically a tangible increase in revenues from markets requiring localization, or an increase in traffic to a Web site from visitors now able to browse in their native language. In institutional or government contexts, return on investment in localization could be characterized by success of a given project in a given region, or by a marked increase in the quantity or quality of intelligence from a given region.

With increased opportunities — commercial or other — based on localized content, the scope of localization technology has evolved in the last decade (and continues to evolve) from relatively small point solutions to large-scale platforms, or enterprise business solutions. Localization point solutions range from commercially-available desktop workstation applications designed to increase translator productivity, to proprietary utilities in the form of scripts, macros and small applications designed to perform specific production tasks such as pre- and post-processing of complex content types. Most point solutions are inexpensive to purchase and maintain, but deliver only modest return on investment, because of their inability to scale to very high-volumes of content or to address more than a limited number of discrete tasks in complex localization processes. Pre- and post-processing workflows for complex content types can be intensive (see Savourel 2001: 363, 375), creating the need for greater automation and more sophisticated software solutions to achieve economies of scale in enterprise localization scenarios.

The evolution of localization technology from relatively inadequate point solutions to enterprise localization platform solutions was catalyzed by the explosive growth of the Internet as a commercial medium in the mid to late 1990s, "enabling anywhere, anytime service in a great variety of fields" (Korine and Gomez 2002: 66) and making globalization a business imperative for companies with an e-commerce focus. Conventional localization technology point solutions could not deliver adequate time and cost savings to companies making significant investment in content globalization. With an increased drive for global product and service delivery, localization was more visible as a tool in the critical push for global markets and global branding, allowing companies to "capitalize on markets that years ago would have been unthinkable without foreign offices" (Yunker 2002: 80). The increased commercial significance of localization coincided with advances in information technology in the late 1990s, specifically the advent of multi-tiered Web applications (Cheng 2002: 35), driving the need for more sophisticated localization technology solutions to keep pace with the demands of Web-based content management and delivery.

Despite the slowdown in e-commerce investment following the demise of the Internet bubble, greater demand for digital content management and publishing solutions has resulted in greater visibility for localization in the enterprise business application environment. The value proposition of localization technology has therefore shifted its focus from enhancing translator productivity and reducing translation service provider costs to facilitating global information management for multinational corporations, organizations and government entities. Evidence of this is the creation of the Globalization Management System (GMS) market segment (Shadbolt 2001: 41).

The GMS is the first localization technology actually marketed as an enterprise business application, and the first to promote the use of formal ROI tools both during the sales cycle and throughout implementation. The GMS is fundamentally a multilingual content management solution and a business process automation solution, but with the integration of analytics and data mining tools a GMS can also be used as an important business intelligence tool to measure the effectiveness of a given implemented localization strategy.

Value proposition

As discussed above, GMS technology was originally developed to address the challenges of localizing complex Web sites in the heat of the Internet boom, when the Web became the "standard publishing platform for global business" (EEEL Xerox Case Study 2003: 2). Large companies and organizations were suddenly committed to a global Web presence and faced unique content strategy challenges. Global content deployments imposed significantly more complex business, legal and technical infrastructure requirements than single-language/single-market deployments. Companies not prepared for the investment required to establish and maintain a global content strategy often found they had to scale back features, services and market presence as they deployed multilingual sites and intranets, thereby losing significant competitive advantage and prior investment.

One key value proposition of the GMS was therefore to make Web-based content localization easier by streamlining processes, reducing costs and allowing companies to roll out e-commerce services faster to foreign markets via a multilingual Web site (Shadbolt 2002: 5). Another key GMS value proposition has been to strengthen global brands for corporations that maintain a significant Web presence in multinational markets. With the establishment of the Web as a powerful marketing medium, the need to publish fresh content with local relevance while consistently representing core branding and corporate messaging across markets remains an ambitious goal.

While Web localization is still a core GMS value proposition, the content management market has changed since the Internet boom-and-bust period, and GMS technology has evolved into a mature supply-chain management solution, designed to streamline production of any type of multilingual content. Data from some early GMS or hybrid GMS-CMS deployments has also become available, providing insight and metrics useful to current and future deployments.

Best practices and formal measurement

Metrics are critical to building credible business cases (Keen and Digrius 2003: 46, 63–64), as are best-practice models that guide the generation and collection of useful metrics. A number of formal efforts are underway to define methods for identification and collection of localization-related data points, including the Localization Metrics Initiative (LMI), specifically for client-side data (Schäler 2003: 11), the nascent GILT Metrics standard (see http://www.lisa.org/oscar), and the Excellence in European eContent Localisation (EEEL) project.

Running for 18 months from 2002 to 2003 and Funded by the European Commission, the EEEL Project (see http://www.eeel-online.com) developed the eContent Localisation Maturity Assessment (ELMA) model, a framework to formulate high-level goals in content globalization as well as a tool to score an organization's readiness/effectiveness relative to industry benchmarks for these goals. This project represents a tangible effort to give structure to the business case for localization by providing seven topic areas that map up to high-level content publishing goals and map down to specific issues and quantifiable maturity statements. Technology is one of the seven topic goals, with multilingual content management as a prominent issue area.

Three case studies from the project are analyzed here to illustrate the role of GMS (or GMS/CMS) technology in the efforts of companies to maximize the efficiency of their globalization efforts and generate positive ROI. While these studies were not benchmarked quantitatively using the ELMA model, they nonetheless give qualitative insight into current industry practices and provide background for assumptions and conclusions offered in this article.

EEEL case study: Xerox

The Xerox case study describes a carefully planned project aimed at addressing global brand dilution and Web publishing process overhead identified by the company as the result of a decentralized global Web content model. The project's primary objectives were to grow business-to-business and business-to-consumer revenue opportunities through Xerox.com, while reducing the overall costs of maintaining a more centralized and technologically sophisticated global Web strategy. By putting in place an infrastructure that supported multiple locales from the beginning, the company hoped to promote the global Xerox brand while effectively servicing local markets with the same attention to local requirements that the former decentralized model offered. Aside from increasing business opportunities and brand recognition, a key objective was also to reduce the overhead of maintaining a global e-commerce infrastructure.

GMS technology was developed internally to automate and manage translation processes. Key to the strategy of building a global infrastructure with local awareness was storing content centrally with tags for language and geography. This facilitated re-use of common content while allowing for locale-specific customization. This approach is discussed in detail as a "hybrid model" in the context of content taxonomy creation by Nguyen et al. (2002: 10).

The conclusion of the study is that process improvements were definitely made as the result of implementing the Xerox.com content strategy. The company overall has benefited from the combination of centralized control over certain site content on one hand, and a flexible framework to meet the needs of regional sites on the other hand. Thus the business objectives of increasing the effectiveness and production efficiency of Xerox.com were achieved. The study also offers an example of the difficulty involved in determining return on investment for centralized content deployment strategies with country-level localized sites. Each country site is funded by the local company branch, which is most likely its own profit and loss center. Therefore the ROI of each individual country site can really only be determined by the cost center responsible for it. This inability to collect global metrics on investment in the overall site rollout is flagged in the study as an information management action item for the company.

EEEL case study: Fluke Networks

The EEEL Fluke Networks case study illustrates how a corporate content strategy based on globalization drove investment in technology to help grow and enforce the strategy. The study indicates that Fluke Networks correlated the quality of its localized Web presence to the success of its products in given markets, and that brand recognition was one of its key assets.

The company implemented a centralized content authoring strategy (English as source) and localized its site to different depths (i.e., more localization or less localization) according to local language requirements. However, in the case of Fluke Networks, no GMS was implemented, all content localization being handed off manually or semi-automatically to a localization vendor and then returned to a CMS for deployment.

The study indicates that Fluke Networks employed no formal financial measurements to assess whether investment in its localization strategy — including technology — was returning value to the company. However, the company does state in the study that its marketing strategy, customer outreach efforts and time-to-market of product information all benefited qualitatively from addressing non-English market segments via its Web globalization efforts.

EEEL case study: Bankinter

Addressing the business issue of multilingual marketing within a single national geography, the EEEL Bankinter case study offers a look at a traditional service sector company using technology and language to increase market share. Bankinter is described in the study as a telephone banking and online banking pioneer in

Spain, differentiating itself in the banking services sector by its early adoption of remote service delivery to the extent that at the time of the study more than 60% of transactions were done remotely, with 40% taking place via the Internet. The primary globalization drivers were extending service to non Spanish-speaking clients, as well as growing the online business-to-consumer channel.

Bankinter's complex Web application infrastructure (including a database and CMS) led to the decision to implement a GMS for process management and control. The GMS allowed for rapid content localization changes, reduced the number of human resources required to manage the process, reduced the complexity of the language translation and review process, and eliminated the risk of inadvertent damage to non-translatable application code.

The study describes the implementation process and the substantial internationalization groundwork necessary to support multiple locales within Bankinter's online architecture. Unlike the Xerox case study, in which technology was adapted to existing process, Bankinter opted to adapt its localization process to the GMS technology solution it adopted.

The study indicates that ROI was assessed only intangibly by online traffic statistics (page hits on localized portions of the site) and internal stakeholder feedback. The study concludes that the globalization initiative — while limited to a single foreign locale — provided Bankinter with the technical and organizational foundation for future localization efforts.

Conclusion

Gauging return on investment in GMS deployments — whether based on commercially-available technology solutions or solutions developed in-house — is complicated by the range of other investment factors typically involved in GMS rollouts. As was seen in the EEEL Xerox case study, globalization technology and process investments were made in the context of a larger global Web initiative, and ROI for the project overall was difficult to quantify outside of the financial context of each country-based cost center. An obstacle to cost/benefit calculation, this issue is compounded by the fact that investment levels, implementation costs and corresponding revenue increases are most likely subject to different accounting rules in multinational deployments, and therefore are difficult to factor into a larger ROI calculation. This illustrates the fact that fundamentally "ROI is different in different segments for investment in identical assets…" (Friedlob and Plewa 1996: 39).

Overall, however, the lack of tangible data in the case studies cited is most likely due to improper or inadequate measurement techniques, and not to the quality or availability of the data in each case. Collection and analysis of relevant data require a level of investment that was perhaps not available to these case study participants. To facilitate collection of relevant data, the ROI of GMS implementations could be separated from the overall success or failure of a localization business case, in the context of a global content deployment effort. If ROI measurements focus more on the core GMS value proposition (time, cost, quality) as opposed to the overall benefit of — for example — a global site re-deployment, it could be easier to determine how turnaround, cost and quality of localized deliverables are affected by a GMS implementation. This data could then stand on its own and be made available for subsequent inclusion in an overall calculation of ROI for a given marketing campaign, etc., but at the very least the degree to which the GMS achieved its stated goals would be quantifiable.

Nevertheless, despite the difficulties expressed in the EEEL case studies in determining the tangible benefits of globalization technology solutions, the intangible or so-called "soft benefits" (Keen and Digrius 2003: 274) were clearly demonstrated. As existing GMS deployments mature, as best practices models are adopted and as investment levels in GMS technology increase, so too should the number of published case studies documenting demonstrable ROI.

References

Cheng, S.-C. 2002. "Multilingual integration of GMS and content systems." *MultiLingual Computing & Technology* 13 (5): 35–38.

DePalma, D. 2002. *Business without Borders: A Strategic Guide to Global Marketing*. New York: John Wiley & Sons.

———. 2003. "Rage against the content management machine." In *LRC '03: The 8th Annual Localisation Conference and Industry Showcase*, 69–77. Limerick, Ireland: Localisation Resource Centre.

Excellence in European eContent Localization. 2003. "EEEL Xerox.com Case Study." http://www.eeel-online.com/app/eeeldes.nsf/eeelookup/286BA84AE839AB9DC1256D3C00366831/$file/EEEL%20Xerox%20Case%20Study.pdf

———. 2003. "EEEL Fluke Networks Case Study." http://www.eeel-online.com/app/eeeldes.nsf/eeelookup/4EA983E01E67B147C1256DA4003D7F5E/$file/EEEL%20Fluke%20Case%20Study.pdf

———. 2003. "EEEL Bankinter Case Study." http://www.eeel-online.com/app/eeeldes.nsf/eeelookup/BC0FD22D145B7343C1256DC6004177AD/$file/EEEL_EBankInter.pdf

Friedlob, G. and Plewa, F. 1996. *Understanding Return on Investment: Getting to the Bottom of Your Bottom Line*. New York: John Wiley & Sons.

Keen, J. and Digrius, B. 2003. *Making Technology Investments Profitable: ROI Road Map to Better Business Cases*. Hoboken: John Wiley & Sons.

Korine, H. and Gomez, P.-Y. 2002. *The Leap to Globalization: Creating New Value from Business Without Borders*. San Francisco: Jossey-Bass.

Nguyen, T., Flynn, S. and Girimohan, C. 2002. "Global content management systems." *The Guide to Content Management.* Supplement to *MultiLingual Computing & Technology* 13 (1): 8–13.

Savourel, Y. 2001. *XML Internationalization and Localization*. Indianapolis: SAMS.

Schäler, R. 2003. "Making a business case for localisation." In *Translating and the Computer 25: Proceedings of the Twenty-fifth International Conference on Translating and the Computer*, N.p. London: Aslib/IMI.

Shadbolt, D. 2001. "Meeting the challenge of globalization management." *MultiLingual Computing & Technology* 12 (8): 41–44.

———. 2002. "Options for managing content." *The Guide to Multilingual Content Management*. Supplement to *MultiLingual Computing & Technology* 13 (1): 4–7.

Wagner, E. 2002. "Steps to creating a content strategy for your organization." *The eLearning Developers' Journal* (October 29): 1–8.

Yunker, J. 2002. *Beyond Borders: Web Globalization Strategies*. Indianapolis: New Riders.

Localization Cost*

Carla DiFranco

The localization industry spends much time and energy examining cost and attempting to contain it. Localization service providers (referred to in this article as localization vendors) and their clients often begin to examine project cost as a reactive endeavor, only after the technical complexity and number of hours spent on a project begin to spiral out of control. However, taking a proactive approach to project cost before problems arise is both easier and a far more effective way to save time and money. Viewing the project from 10,000 feet and then taking a closer look at specific stages in the process will enable any project owner to identify and understand the various factors that impact cost. This viewpoint will also make it clear to the project owner that most of the fundamental localization cost drivers are in fact upstream (client-side) issues that fuel downstream (vendor-side) costs. This article will focus on some of those upstream issues. Understanding the nature of costs in localization projects, and the reasons why certain projects suffer budget overruns, is not an easy undertaking. It is a given that all localization services will cost money. All or most of the content that the software user sees must be translated or localized into one or many languages. Translators and localizers, who are essentially the backbone of this industry, must be compensated for their work. Language service costs in this sense are merely a cost of doing business in the international arena. However, wise management of these language resources as well as the tools used during the localization process can make it easier to plan for localization cost proactively and effectively — especially when facing a limited budget or lack of understanding about the importance of language quality in the final product.

This article will draw upon the author's experience and explore some of the ways in which localization costs can proactively be controlled. The ideas proposed in this article are not exhaustive. To simplify the discussion, the larger issues have been condensed into three relatively broad groups: tools, process management and globalization. Within each category, issues are identified and possible solutions provided. The order of these groupings is not random: this article follows the localization food chain, beginning with the individuals who perform the actual

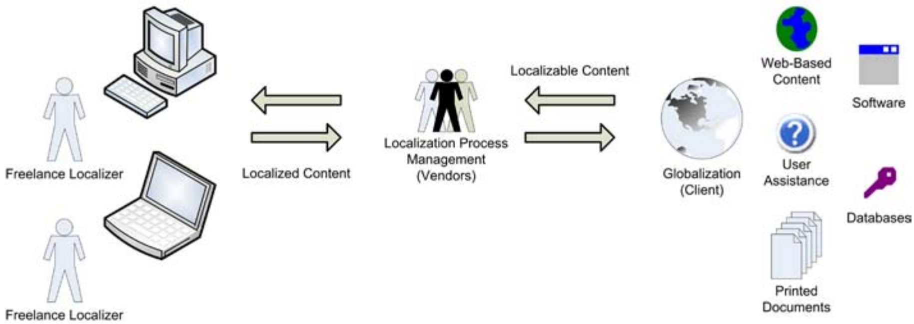

Figure 1. Localization process, from individual subcontractors to the source of localizable content.

localization tasks (for the purposes of this article, freelance localizers or single-language localization vendors), then moves upstream to larger localization vendors (multilingual vendors and agencies), and then finally to the source of the localizable content, namely the client (see Figure 1).

This order is intended to underscore the fact that much of the cost incurred during localization is the result of practices that take place upstream, long before the ostensible localization project actually begins. It is important to note that many of the examples listed here are project-dependent, and it is the hope of the author that readers can draw upon important points for use in similar project types.

I. Tools

Using the most appropriate localization tool and limiting tool proliferation during a project saves time and money in the long run.

The term "tool" is used in this article to refer to any specialized piece of software used in the localization cycle. CAT (i.e., computer-assisted translation) tools are widely used in localization due to the prevalence of repetitive content such as user interface strings and help topics that may be used over and over again throughout a product or a documentation set. CAT tools enable users to translate these content types directly, and to recycle their translations as they work. Localization tools offer similar functionality and also allow users to solve basic engineering issues. For example, recurring help text can be recycled throughout a documentation set using a CAT tool, thus ensuring the consistency of the localized content, and dialog boxes and their controls can be resized using a localization tool to accommodate translation-related expansion in Finnish, German, or even Indic languages that typically take up 40% more space than English, thus ensuring the correct display of all strings in the localized user interface (see Figure 2).

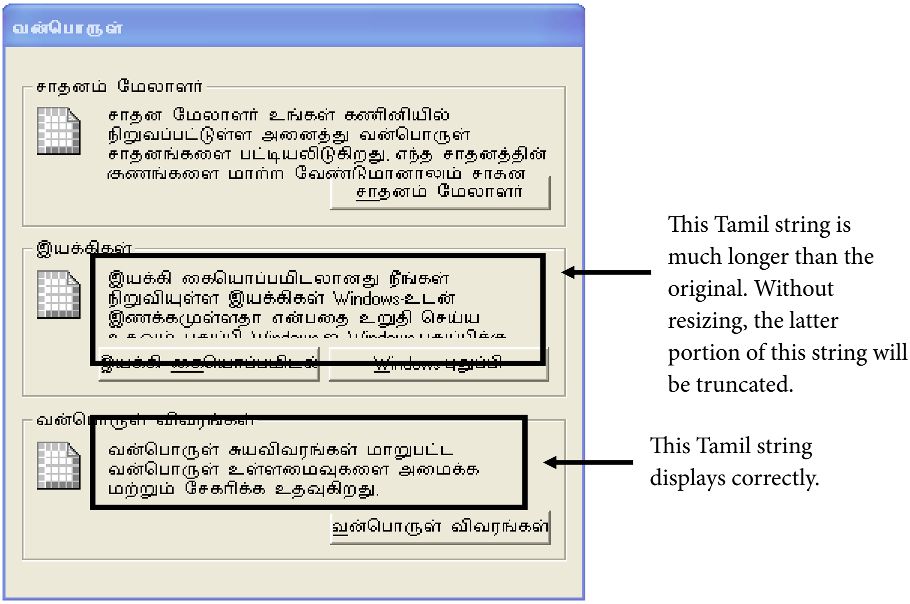

Figure 2. This dialog box must be resized to accommodate expansion and thus avoid truncation of the localized Tamil strings.

A number of tools on the market today enable both translation and localization engineering tasks. Two basic features of these tools are important to mention here:

1. Most tools are designed so that the user can see or manipulate localizable content in context, such as in the help documentation or in the user interface in which it appears. (However, it should be noted that not all content is visible to the end user. For instance, taxonomies, indices and keywords may require translation to ensure search functionality in a localized product, but such elements are not visible *per se* to the end user.) Localization tools allow the user to translate, edit, resize controls, and so forth. Settings in the tools enable the user to parse source files and separate localizable content from non-localizable content. In Figure 3, a localizer will only have access to some inline HTML tags that provide formatting. The <B> tag, for example, is an inline formatting tag that tells the browser to display the corresponding text in bold type. As such, the location of the tag may need to be changed in the localized sentence. Other tags such as <LI>, which tell the browser to display the corresponding content in an ordered list format, should not be touched by the user.
2. A translation memory is a database that records the translation work performed on any content so that it can be reused (recycled) on the fly. Some translation tools also include a built-in machine translation component.[1]

Figure 3. In this example, tags are parsed and automatically protected from accidental change or deletion. This tool uses colors to distinguish different tag types.

It is relatively easy to save money using translation tools provided that the tool chosen is appropriate for a given job and/or organization, and provided that users and project managers understand how to use the tool effectively. However, a quick survey of the various computer-assisted translation tools available today suggests that finding the right translation tool is not always an easy endeavor. Determining which tool is right for the job requires that the user consider a number of factors:

- Scalability: Can the tool handle a variety of project types (large, small, complex, all required language pairs)?
- Workflow: Related to scalability, how does the tool recycle content? Does it record tagging and/or formatting in the TM? Does the tool allow individual subcontractors working at multiple offsite locations to merge their translation memories to ensure consistency across the project? (It is important that any tool provide a convenient method for updating translation memories as an ongoing part of the project workflow.)
- Compatibility: Can the TM be reused by/in other tools with minimal loss of data? Some localization tools produce output that cannot be used with other localization tools, and there is always a risk that quite a bit of recycling may be lost if content is recycled using different translation tools (Zerfass 2004). Are clients, vendors, and/or localizers using specific localization tools? If so, which ones? The tools used by other participants in the localization cycle may affect the decision as to which tool(s) to adopt.
- Customization: Does the tool allow further customization if needed? Some localization tools provide open APIs that allow automation where necessary.[2] Open APIs enable the potential automation of localization processes that would otherwise have to be performed manually. Depending on the level of customization needed, and with the proper resources, it may be cost-effective to customize a tool in house to suit the needs of a particular project.
- Support: What forms of technical support are offered for this tool? In other words, is official support available from the company that creates the tool,

and/or are there online user groups or other resources that can be used for support? Will support be available and forthcoming when things go wrong? If so, what form(s) will such support take? Is a fully functional version of the tool available for a limited trial period to test its appropriateness for a particular project or file type?

- Ease of use: Is the learning curve gentle or fairly steep? Is training available, and if so, at what cost? If a tool is so complex that lengthy training is necessary for rudimentary operations, then the total cost of ownership may ultimately prove prohibitive.
- Multilingual support: Does the tool work well in all languages? Complex scripts and Unicode-only languages are a great way to test the reliability of a tool. It is important to ascertain whether a tool can handle all languages into which localization or translation is currently performed, as well as those that may need to be supported in the future (even though it is often difficult to know what languages or markets clients may be interested in). For example, some localization tools store the translation memory in ANSI format only.[3] This means that Unicode-only languages (such as Hindi) cannot be localized using the translation tool without resorting to some type of fix or codepage hack. Unicode-only languages are those languages for which no ANSI codepage exists. If such a tool is used, when a multilingual project comes along that includes Hindi, it will cost many more project management hours and training time to manage and support a split process as the product is localized using two (or more) different tools.
- Version control: Does the tool offer file management features? When the translator/localizer deals with updates during the course of a project, a good file management process is critical to avoid performing work more than once or delivering the wrong version of a file. Also, it is important that valuable project management hours not be squandered by tracking files manually.
- Terminology management: Does the tool allow easy flagging and updating of terminology during the project? It is important to not underestimate the value of this feature: most software localization projects will entail last-minute terminology changes because the software code and user interface strings are rarely frozen until shortly before the product is ready to be released to the market.[4]

These questions provide a good starting point for researching the appropriate translation tool for a given use. Purchase price and product-specific support costs of CAT tools are not mentioned in this article because these costs must be evaluated in the context of total cost of ownership, i.e., by considering not only the extent of savings that can or may be achieved by using the tool, but also the learning curve, reliability, relative recycling precision and amount of troubleshooting associated with the tool. For example, some tools that have a modest purchase price may prove more costly in the long run, as more time is spent troubleshooting or duplicating work. After choosing the appropriate tool for a given project, it is important to develop and follow robust localization process management strategies, which help ensure that cost savings and increased productivity are actually achieved (see Section II).

It is also important to note that not all tools used in a localization project are actually "localization" tools. In fact, many, such as compiling, testing and QA tools, play no direct role in localization or translation of content. Ironically, however — or perhaps not, depending on one's perspective — the use of such tools often proves to be especially problematic. due to the fact that they are designed not for software *localization* but rather for software *engineering*. On the other hand, these utilities are often designed in English, for English. Therein lies the problem: regardless of how often the phrase "English is just another language" may be used, this sort of reductionist thinking generally causes problems when it comes to many of the specialized software engineering utilities used in localization, due to the problems they cause when working in languages other than English.

Many issues can arise when a localization team uses specialized tools to rebuild components in languages other than English. For example, if a help compiler is used for a project that cannot accommodate Unicode-only languages (such as Hindi), it will be impossible to compile help files localized into such languages. A multilingual localization project that includes compiled help will thus require a separate process or help type for Unicode-only languages. Tools that offer inadequate support for one language or another are a primary reason why new tools are generated to accomplish just one task. Such tools may be quickly developed for a given project with little foresight, used on that project, and then set aside — perhaps never to be used again. The temptation to create dedicated utilities can be strong and tool proliferation can easily spiral out of control.

Overall cost savings on a localization project tend to be inversely proportional to the number of tools in the project arsenal. Tools originally created to save time by automating processes may end up actually costing more time and money in the long run. For example, the creation of 20 or more small tools during a longer localization project will increase the overall amount of time and effort spent on roll-out, training and managing. Likewise, the greater the number of tools, the greater the amount of time and money that must be spent to determine appropriate processes for the various language/tool combinations and to differentiate localization bugs from the new bugs produced by these tools. Last but certainly not least, return on investment is marginal at best if the usefulness of these tools does not outlive the current project. Creating quick fixes to immediate problems is inevitable in some cases. However, when tool-related problems are addressed upstream, tool proliferation and the costs that they entail can be effectively mitigated (see Section III).

A sample software localization project will serve to illustrate the problems associated with insufficiently globalized tools. This project contained 400,000 words in the software user interface and 2 million words in the user assistance materials (i.e., Help). This project was localized into German, French, Italian, Spanish, Brazilian Portuguese, Portuguese, Danish, Norwegian, Dutch, Swedish, Finnish, Czech, Hungarian, Arabic, Hebrew, Greek, Russian, Japanese, Korean, Simplified Chinese and Traditional Chinese. All the tools used to author, build and test the (English-language) content were optimized for US English. After the project began, the localization team discovered that these tools only supported Western European languages. This forced the localization team to make a decision to either overhaul the entire process for all languages and request that the build and test tools be reengineered for localization, or use the US English-specific tools for the project's 12 Western European languages and create new tools to serve the needs of the other languages. Either solution would have turned out to be a costly one, simply because the correct tools were not created from the outset. The first solution would have been less costly in the long run because it decreased the future likelihood of such problems. After all, if the onus was on the developers to fix their mistakes, it is a good bet that this team would not want to repeat them the next time around!

As it turned out, the second solution was implemented in this process. New build and test tools were rigged together at the last minute to be used on all the other languages in the project. This resulted in the use of over 30 different tools, some of which performed the same function as the originals but in different languages. Localization service providers had to quickly learn how to use all these new tools before the project started, which took valuable time away from other important endeavors, such as project planning. A greater number of errors made their way into the final build because time pressures did not allow for optimal training in the use of these new tools. This increased testing time forced many team members to work overtime to create the final builds. This nightmare scenario could have been avoided if the tool developers had consulted with the localization team during the planning process to ensure that the build and test tools worked for all the languages in the project.

As this example shows, tool proliferation is often the result of a knee-jerk reaction to problems that are discovered at the last minute. Why do those who work in localization spend time putting out fires instead of preventing them? Project schedules continue to shrink as the amount of content to localize and the technical complexity of localization projects continue to expand.[5] The average localization schedule generally reflects this trend: although it can take 2–3 years to develop a large product in English, only 6 months may be allotted to localize the product into 20 or more languages. The number of tools required for successful localization can only be reduced if the content to be localized does not require the use of so many different tools to build, test and perform other important project operations in different languages. Ensuring that the product can be built for all languages using the same core set of tools serves to streamline the number of tools used on the project, and offers a concrete strategy to help contain localization costs over the long term.

II. Process management

Localization cycle processes and process management can also be simplified to save time and money. A generic localization project can be used as an example: processes can be grouped into four major categories. The processes below assume a client-vendor-freelancer relationship in which localizable content is passed from client to vendor to freelancer, and the corresponding localized content is handed back from freelancer to vendor and finally to the client (see Figure 1). Specific roles have not been assigned below as they may vary depending on the given situation. For each of these general processes, certain proactive strategies are suggested to reduce costs.

1. **Project planning.** During the planning phase, target languages are chosen, as is the extent of localization for different markets. Legacy TMs or legacy materials for the project are gathered, project recycling (per-word cost) is assessed from the legacy TMs and materials, and old project post-mortems (if any) are reviewed. This last step should provide a good overview of the obstacles that were encountered the last time around, and thus a good idea of what problems should be proactively addressed this time around prior to project launch. Additionally, localization vendors are chosen based on a number of factors, including the ability to handle requisite volumes, quality produced in the past, ability to use project-specific localization tools, and so forth.

- Legacy TMs should be checked to see how much of the new content allows recycling using previously localized content. If the new project content corresponds to more than one legacy project, it may be a good idea to experiment with combinations of different TMs from past projects to see which combination enables maximum recycling, or ensure that certain TMs are used for reference only. Last but not least, it is also a good idea to ensure that duplicate translations in the TM are checked, and if necessary, to create guidelines for localizers.

- The decision to fully localize products should be based on demonstrated user needs and preferences. In some cases, partial localization of key features may be an option. It pays to do research ahead of time to determine the appropriate extent of localization.

2. **Localization.** During the localization phase of the project, it is important to ensure that a proper schedule is maintained. Localizable files should be handed off according to the schedule, and use of a good file transfer mechanism is a key element to the success of this process. Because product development is rarely complete when localization begins, these two processes will likely overlap at some point. Consequently, it is important that the localization, development and writing teams work together to identify the sequence in which the various chunks of content will be finalized. Using this information, localizable content can be grouped into batches and scheduled for sequential hand-off to a localization vendor or freelancer. This type of planning should also include time for last-minute changes to the product that occur after regularly scheduled batches have been dispatched. Finally, localization tools and processes should work correctly for localization vendors, and terminology queries should be answered promptly.

- Because final software code freezes typically occur just prior to the product's release to market, most projects will require updates to files that have already been handed off to localization vendors to work on. How these changes are handled will determine whether or not the correct files are ultimately built into the final product. Changes should be controlled; depending on the complexity of the project, it may be less confusing for localization vendors if changes are received in batches rather than one file at a time. Advance notification of such updates is always helpful, and often it is a good idea to schedule batches of changes as part of the initial project plan.
- Localization vendors should know who they can contact for support when problems are encountered with any of the tools used in the project (whether such tools are commercial or proprietary). If there is enough time in the product cycle, it is always a good idea to provide sample files to test the full process. If the tools or localization vendor are new, it is advisable to run a mock process to uncover (and ideally, resolve) potential issues from file transfer problems to incorrect localization tool settings and poorly globalized[6] build and test tools.
- Source- and target-language terminology often raises issues during localization projects. The use of inconsistent terminology in the source materials will result in terminology queries from localizers and vendors, and thus in increased

cost, due not only to the time spent fielding, addressing and managing queries, but also to the loss of potential 100% matches in the translation memory. Conversely, if the software and user assistance are localized at the same time by different vendors, or by the same localization vendor but in different locations, target-language terminological consistency becomes a potential issue. In such cases, it is important to ensure that someone is available for terminology assistance in case there are questions. Localization vendors and freelancers will end up squandering valuable time researching terminology if resources are not provided. Similarly, if a localization vendor or freelancer is using incorrect target-language terminology from the start of the project, it will be more costly in the long run to go back over everything and fix errors after the localization project has been completed. Of course, any extra hours spent on the project for terminological research as a result of inconsistent source terminology are charged to the client (and rightfully so). However, streamlining the process to ensure that actionable replies to queries are provided promptly and made available to all vendors involved in the project allows better control of the time spent on terminological queries. It is also important to keep in mind that terminology changes are often made late in the project cycle. It is possible that a marketing department may change the product name (this happens more often than we would all like to admit), or there may be last-minute changes in target-language terminology as the user interface is locked down. All of these changes jeopardize final deadlines. Padding the project schedule as much as possible to accommodate these types of contingencies is a good strategy if the time can be afforded at the outset.

3. **Test, QA, build, build checks, check-in**. For this phase of the project to be successful, it is important that all test and build tools properly support all languages, and that localization bugs be called to the attention of the appropriate party (or parties) to be fixed in the target files and in the translation memory for each language.

- Similar to the mock handoff to a localization vendor described in the previous section, it is always a good idea to perform a pseudo-localization on the project content and then follow every single process that a localization vendor or freelancer will also perform.[7] In effect, this simulated project run-through can serve as a stress test on a number of project processes including translation tool settings, test tools, test process and build tools. Following this dry run all the way through to build ensures that all localization instructions are correct. If localization vendors have an opportunity to run though the entire process once, any problems encountered can be addressed as they occur during the simulation. It is then unlikely that these problems will arise during the actual project; but if they do, the proper solutions will be known. Not every project will require a dry run of this type. However this method is especially useful when new processes or file types are introduced.

- Before testing begins on the software and help content, a good testing plan should be in place. This may sound a bit trite, but it is important that the correct tests be performed by the correct person in order to avoid duplication of effort and to save time and money on the project. For example, it makes sense to assume that localization vendors are fully responsible for linguistic testing since the client may not possess the requisite language expertise to perform such tests. However, functional testing is a completely different story. If localization vendors receive vague instructions, they may simply go ahead and perform such testing, assuming that it is part of their responsibilities. At the same time, if the (client) software development team is also performing functional testing, duplicate bugs may be logged. Tracking and resolving duplicate bug logs can quickly become a logistical nightmare. This issue can be solved by ensuring that testing responsibilities are clearly defined and documented at the start of the project.
- When delineating project responsibilities, it is advisable to also clearly define who will be responsible for fixing bugs. The severity of source-language bugs should be determined by a pre-defined triage process. Triage helps distinguish localization bugs from functional bugs in the core product, thus making it easier to ascertain who should fix them. As a general rule, any bugs that are introduced during the localization process should be fixed by the party that introduces these bugs, at no charge to the client. Conversely, it is incumbent upon the client to fix any bugs that are present in the core product. If bugs are found in the source-language product by a localization vendor or freelancer, it is appropriate that they be brought to the attention of the client. Ideally, a triage process that is clearly defined and communicated to all project participants helps ensure that that everyone understands which bugs should be reported, how, and to whom.
- Often the bug fixing phase is a rushed affair at the tail end of the project, after most of the localization is complete. In such cases the localized deliverable may be handed back to the client at the last minute. Meeting deadlines is always difficult if problems arise. Nevertheless, it is important that attention also be paid to the translation memory at this stage. At the end of any project, the translation memory deliverable should accurately reflect the final product deliverable. In other words, the content of the TM should match that of the definitive product files as completely as possible. It is advisable to check TMs after the project is complete either by performing a full analysis on all project files or by testing a random subset thereof. Any acceptable quality thresh-

old should allow for a slight margin of error, but ensure that the TM content matches that of the localized product. CAT tools that offer an open API can be automated to perform this checking process recursively throughout a set of languages.

4. **Post-mortem, cost analysis, TM linguistic checks, archiving.** At the end of the project, all parties should participate and have their say in the post-mortem. Any issues encountered during the project should be documented in the post-mortem so that they can be addressed, and ideally, avoided altogether the next time around. This post-mortem should then be archived with the final version of the files, the translation memory and any other necessary materials that would assist future project owners.

- Before archiving is done, it is always advisable to perform a functional check on the TM (see Section 3). The linguistic quality of the TM is equally important, but is often much more difficult and costly to check. Nevertheless, if quality is not maintained in a project TM, the effects of this poor linguistic quality may follow the product from version to version. For example, if localization vendor X was used for version 1.0 and the client switches to localization vendor Y for version 2.0, vendor Y may express reservations about the quality of the TM and whether it can or should be used for version 2.0. Such situations arise in projects every now and again, and each time, the vendor may insist on editing the content in the TM to correct mistakes from the previous vendor (whether real or perceived), which costs time and hours up front before the project even begins, or ends up costing valuable time after the project has already started. A well-established linguistic quality control process on the final TM deliverables, performed by a disinterested third party, is vital to avoiding such situations.

The processes that comprise a localization project interconnect like the pieces of a puzzle, and the different tasks should be clearly defined with regard to all participants. Clear delineation of expectations and responsibilities is important in any large project, but it is often absolutely critical in a client-vendor-freelancer scenario (see Figure 1). Often these parties operate in three or more different locations around the world. Not only is it often impossible to drive across town for a meeting, but time zone and cultural differences may also affect communication. To ensure that the project phases outlined above are completed smoothly, contracts that define roles and deliverables must be established at the start of any project. Schedules should be adhered to as closely as possible, and updates should be provided to all parties concerned.

Even when project expectations are clearly defined, issues will often arise. For example, a project manager may change roles, engineers may leave the project, freelancers may not be available for the full project, or workloads may be split. All of these changes necessitate adjusting or re-evaluating the process and/or the schedule. For example, suppose that a client arranges training for the localization vendor as part of a large project kickoff. The localization vendor, in turn, sends project managers and engineers to be trained at this kickoff. If the project is subsequently delayed by 6 months or a year, it is likely that the people who have completed the training will be working on other projects by the time the project actually starts. Even with the best intentions and best planning, some things are simply out of the project owner's control.

Project delays that undermine training efforts are a frequent occurrence. Rather than ferrying a small team of people halfway around the world for training that they may never use, it is cheaper, and more effective in the long run, to develop training materials that can be distributed on CD-ROM to any localization vendor or freelancer. In this way, training is provided to those people who actually need it, when they need it, and less expense is incurred. In addition, incremental training updates can be provided if processes or tools change.

Identifying the ways in which people work well together and ensuring that the process tasks are distributed appropriately can provide substantial benefits when it comes to localization cost. It is important to formulate a comprehensive project plan that minimizes duplication of effort while maximizing automation. Such proactive planning can effectively save both time and money in the long run.

III. Globalization

Any software or user assistance that is to be localized should first be well globalized.

Well-globalized software and user assistance essentially means that the code has been separated from the localizable content, and that the product in question will support all languages without the need for redesign. When materials are properly globalized, the localization process can focus on more specific tasks such as achieving high linguistic quality and consistency (often with the use of CAT tools), ensuring that engineering tasks such as font updates, mirroring[8] and correct dialog box sizing are all performed correctly and that the localized product performs well during testing. The goal of globalization is to eliminate the need to completely re-engineer the product for multiple languages simultaneously. Ideally, globalization ensures that freelancers and localization vendors can focus on their

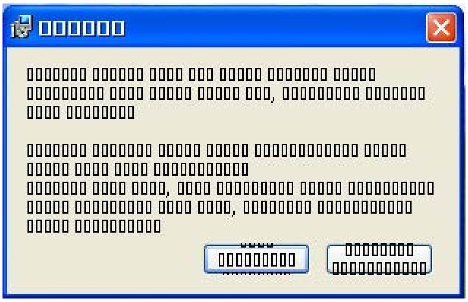

Figure 4. Example of a dialog box that uses hard-coded fonts.

expertise, namely ensuring excellent quality in a fully-functional product for the target markets.

Users are often the victims of poor globalization, especially if a localized product does not work properly. For example, a user may open a dialog box and be unable to read the content because a font used in the dialog box (and hard-coded in the software) does not exist on the user's system (see Figure 4). This renders the message in the dialog completely useless, even if due diligence was given to localizing the text and ensuring that the dialog was sized correctly.

To avoid this and other similar problems, it is important that software products be checked for known globalization issues before localization begins. Although many such issues have over time become the stuff of legends, it would be naive to think that all globalization issues have been relegated to history. There will always be new products and new technology, so there will always be ample opportunities for new globalization issues to arise.

One proactive way to approach such problems is to maintain a checklist of issues that have arisen in the past, and ensure that new projects are checked for each item in the list. Depending on the situation, this list can be used by a localization department or localization team, or it can be provided to the client, with some cost calculations attached, identifying the potential cost of *not* fixing known issues up front. There are a number of ways that the financial impact of incomplete globalization can be communicated, and each situation may necessitate a slightly different tactic for getting this information across.

Another example of poor globalization that appears over and over again in US English products is the sorted glossary or sorted list. Often such lists are sorted in English products, but no methods are provided to localization teams to facilitate sorting in other languages.

Sorting a list of terms is one of the more difficult tasks in localization, as sorting rules are language-dependent, and are very much tied to user expectations. For example, in the screen shot in Figure 5, English terms are listed in English

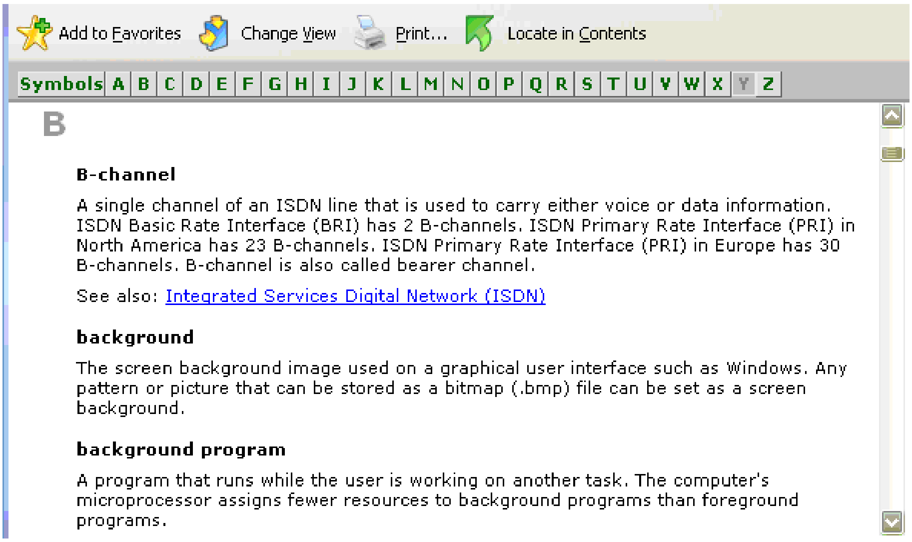

Figure 5. A snapshot of the Windows XP glossary.

alphabetical order, and the full hyperlinked alphabet appears at the top of the file, thus allowing the user to browse for terms alphabetically.

The English list depicted in Figure 5 sorts relatively easily after translation into Western European languages such as German or Swedish, with a few minor adjustments. However, problems arise when different character sets are used, which may sort according to different rules than those by which Western European languages are sorted. Japanese is a particularly complex example. Before sorting the content depicted in Figure 6, an extra term is added to every term in the glossary after localization into Japanese. This extra term is referred to as “ruby text” (W3C 2001). Ruby text is a term that denotes a (kana) pronunciation word in Japanese corresponding to a given Kanji term. In Japanese, the glossary is sorted by way of the ruby text term rather than the actual term used in the glossary and corresponding content (see the glossary terms listed in Figure 5 and the ruby base in Figure 6). This means that although all other languages are sorted based on one same field (the alphabetical term list, i.e., “B-channel,” “background,” “background program,” etc., as depicted in Figure 5), Japanese is sorted based on a completely different field (the ruby text). Ruby text can appear either above or below the Kanji text that it represents. Examples of ruby text are provided in Figures 6 and 7.

Adding ruby text to each term in the glossary is not necessarily an automated process. To a certain extent, ruby text can be automatically added for each known Kanji term. However if new terms are added to a glossary, the ruby text corresponding to these new terms must be checked or manually inserted. The A–Z letter

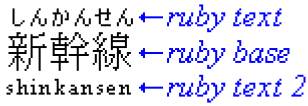

Figure 6. Ruby text as seen by a user (W3C).

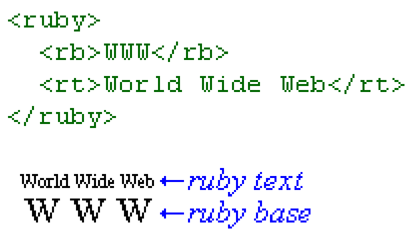

Figure 7. Implementation of ruby text in source code (W3C).

categories must also be replaced in the glossary by the phonetic categories of the Japanese ruby terms in order to allow the user to browse for terms phonetically.

It is important to note that most of the steps listed above require manual intervention on the part of a Japanese localizer. If no automated method is provided to manipulate this content and sort this text, localizers will be forced to manually cut and paste the terms and the corresponding definitions in the file. In the case of small files, this manual work may not require undue effort, but in the case of large files, the engineering time required to enable phonetic-based search functionality will add up very quickly. Time is not the only issue at stake in this process: the risk of errors increases whenever a user manually manipulates code in a file.

A fully globalized, automated tool that enabled sorting of this type of content would eliminate hours of manual work on the part of a localizer or a localization vendor, and would also ensure that the sorting process is carried out in a similar way for each language. Indeed, a broad solution that could be applied to *all* languages would offer many benefits:

1. If there is one process for performing a given task across all languages, then only one process (and one corresponding set of tools) must be documented and supported. Supporting a split process is always much more time consuming and also increases the risk of error.
2. It is easier to create proper test scripts and test cases to check the final output file and avoid introducing errors into the localized build.
3. Initial development of a tool or a utility will prove more cost-effective than paying localizers and localization vendors to perform a given task manually

> during every project. To provide just a basic example, if additional localization costs of $500 per language are incurred for each project that requires manual sorting of a glossary (according to the method described above), and if 20 projects a year are localized, each into 10 languages, the additional localization costs would amount to $100,000 per year. Assuming that development of a single solution across all languages takes one developer 160 hours at $100/hour, the total development cost would be $16,000. Over the course of a year, $84,000 would be saved. Efforts should also be made to ensure that any sorting solution that is developed will work on current and future versions of the glossary. Tools created for this purpose should be updated periodically to maintain compatibility with new formats.

Sorting continues to present obstacles in many target languages, and although automated tools, utilities and even processes are certainly not out of reach, they are also not easy to develop across a broad set of languages. In addition, there are definite costs associated with developing tools to overcome these types of issues. Whether it is worth committing the time and money necessary to develop such a tool must be assessed within the scope of the company and/or the project. Defining the specifications and requirements for overcoming sorting problems across many languages will require time and resources. Developing the actual globalized, automated tool defined by these specifications and requirements will require additional time and resources. The important thing is that the process/tools developed prove to be cost-effective compared to the amount of time (and the related cost) that would be required to perform the task(s) in question manually over a number of projects and across a number of languages. Taking the advice of limiting tool proliferation to heart, it is important that any solution serve a number of projects rather than just one or two.

Conclusion

Tool proliferation, tool confusion and process chaos are the bane of large multi-language localization projects. Often there is a clearly defined need for multiple tools and processes simply because the source content is either impossible or very difficult to localize without complete re-engineering. However, the way in which localizable content is developed and authored is not the only root cause of the problem. Time constraints are also a source of major problems in any localization cycle. Localization projects go over budget not only because the source content may not be well globalized, but also because schedules for localization projects are exponentially shorter than the schedules for creating the source version of the product in the first place. In addition, the twin problems of insufficient globalization and aggressive timelines tend to compound each other. Would shrinking timelines really pose such serious challenges if the product didn't need to be reengineered over and over again for each language? Experience has shown that good proactive globalization practice can really go a long way toward lowering total cost and decreasing project time. If a product did not need to be re-engineered for all non-English languages, more attention could be focused on quality localization and ensuring that the product has fewer bugs. This would result in shorter time frames for testing, and a shorter cycle overall.

Engagement upstream is one important way to combat poorly globalized products. This engagement might entail working closely with development teams to resolve globalization issues in the core product, or working closely with a client (as a localization vendor) to identify issues before the localization cycle begins. Freelance localizers are an extremely valuable source of information in this regard. They are the ones who will encounter the errors and feel the pain of poor globalization with each word that is localized. Feedback from localizers to localization vendors about the quality of the product is critical and, ideally, should be acted upon by the client to improve subsequent versions of the product. Feedback sent up the localization food chain to the client should not be confined to project post-mortems, but should be provided on a rolling basis or at least at regular intervals throughout the project.

Many of the solutions discussed in this article can be used to ensure that this feedback is voiced. Pseudo-localization and mock handoffs are excellent ways to test processes and check globalized content before large sums of money are spent on localization and resources are used to re-engineer products. The earlier testing and dry runs take place, the easier it is to effect change upstream. Feedback along these lines, accompanied by a cost analysis that details how much a project *could* cost if fixes are not made ahead of time, is often an effective tool for ensuring that changes are made.

On the other hand — and this is really the crux of why this information should flow back upstream to upper management — there should be some accountability on the part of those who produce any type of product that will be localized. The way a product is looked at from the moment of conception should be altered. Rather than thinking about all the great and wonderful features that can be added, there should be greater focus on the basic functionalities of the product and how these may help or hinder the worldwide customer. It is conceivable that certain features may not work well in other languages or in other markets. Such questions should be explored during the original product development cycle. Ultimately it is the worldwide customer who should be the focus of global product development, rather that just the English-speaking customer. It's easy to say that "English is just another language," but much harder to put this into practice by developing truly language-neutral development processes. Ultimately, this will not be possible without a shift in focus that makes the development of fully globalized software a top priority. In the end, such a shift in focus will lower localization costs far more effectively than smart tool choices, good planning, communication about issues and globalization checks.

Notes

* The author gratefully acknowledges the assistance of Sören Eberhardt in the research and preparation of graphics for this article.

1. It is important to distinguish between translation memory and machine translation. Translation memory (TM) simply records user inputs and then produces suggestions for new segments based on what is already recorded in the TM (in terms of relative percentage match). Machine translation (MT), on the other hand, involves automated translation from one language into another, with little or no human intervention. Both the TM and MT are merely tools to assist translators and localizers, and not instruments that can replace them.

2. API is an acronym for Application Programming Interface. "An application programming interface (API) is a set of definitions of the ways one piece of computer software communicates with another… One of the primary purposes of an API is to provide a set of commonly-used functions — for example, to draw windows or icons on the screen. Programmers can then take advantage of the API by making use of its functionality, saving them the task of programming everything from scratch (Wikipedia)."

3. ANSI is an acronym for the American National Standards Institute. "The term 'ANSI' as used to signify Windows code pages is a historical reference, but is nowadays a misnomer that continues to persist in the Windows community. The source of this comes from the fact that the Windows code page 1252 was originally based on an ANSI draft — which became International Organization for Standardization (ISO) Standard 8859-1. 'ANSI' applications are usually a reference to non-Unicode or code page-based applications" (Dr. International 2002: 1012).

4. Code freeze refers to the point in time when no further changes are made to the software UI, and therefore no further localization work needs to be performed on tracked changes.

5. Complexity here is a reference to the many different file types, to localizable content that is not clearly separated from non-localizable code, etc., and not to the relative semantic or lexical difficulties of the localizable content itself.

6. For the purposes of this article, the definition of globalization is that given by Dr. International, namely "The process of developing a program core whose features and code design are not solely based on a single language or locale. Instead, their design is developed for the input, display and output of a defined set of Unicode-supported language scripts and data related to specific locales" (1019).

7. Pseudo-localization consists of a simulated localization of a product. More specifically, pseudo-localization entails replacing the source text in software and help files with random characters from another language (which may include accented characters, pictograms, etc.), and expanding strings in software based on benchmark expansion statistics for a given language. For example, translation from English into French or German typically results in approximately 20–30% expansion (Esselink 331). The pseudo-localized product should be built as usual. When properly automated in conjunction with localization tool(s), pseudo-localization effectively tests the globalization of the product, as well as the settings of the localization tool (s).

8. Mirroring is the process by which the user interface is inverted horizontally to simulate localization into a bidirectional language, such as Arabic or Hebrew.

References

Dr. International. 2002. *Developing International Software*. Rev. ed. Redmond, WA: Microsoft Press. Originally published as Kano, N. 1995. *Developing International Software for Windows 95 and Windows NT.* Redmond, WA: Microsoft Press.

Esselink, B. 2000. *A Practical Guide to Localization*. Amsterdam/Philadelphia: John Benjamins.

Unicode. http://www.unicode.org/

W3C (World Wide Web Consortium). 2001. "Ruby annotation. W3C recommendation 31 May 2001." http://www.w3.org/TR/ruby/

Wikipedia. 2005. "Application programming interface." http://en.wikipedia.org/wiki/API

Zerfass, A. "Testing TMX import/export in several translation tools." *MultiLingual Computing & Technology* 15 (9): 51–54.

PART 2

Localization quality

Quality in the real world

Scott Bass

1. Introduction

Regardless of whether you are the buyer of translation and localization services, the agency that provides the service or the translator who transforms the content, your interaction in the “subcontracting triangle” has an immediate impact on quality. It is debatable whether these interactions can be described as “triangular,” since the connection between the client and the translator is mediated by the agency and thus indirect. However, the triangle metaphor is sufficient for the purposes of our discussion since it allows us to represent the flow of information as well as the complex interactions between the three parties that influence quality in every translation and localization effort.

In side “one” of the service triangle, project requirements and materials to be localized flow from the client to the agency. The agency studies both the materials and the requirements and seeks clarification and confirmation about the scope of the project from the client. In side “two,” the agency attempts to secure the participation of a sufficient number of adequately qualified translators to achieve a quality outcome. Side “three” of the triangle represents the indirect interaction between translators and the client, which may take the form of queries about the

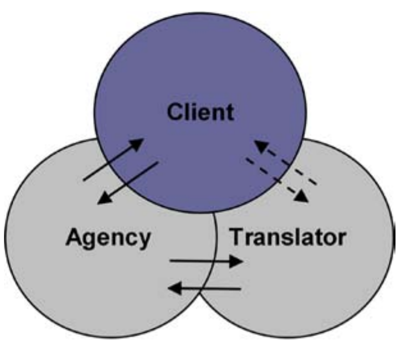

Figure 1. The “subcontracting triangle.”

subject matter. The third side also represents the interaction at the linguistic level by which the translator engages the client via the client's content.

Given the ongoing maturation of the localization industry, why is quality such a burning issue for all in the service triangle? By all indications, quality is receiving more attention than ever. The number of agencies that have achieved ISO 9001:2000 certification continues to increase (but still stands at less than 100 translation companies worldwide[1]). Industry publications have historically paid attention to quality management and to best practices for large localization and translation projects, i.e., those of more than 250,000 words in five or more languages at one time. However, the day-to-day work of small to mid-sized translation companies does not involve projects of this magnitude.

On the contrary, the overwhelming majority of projects are performed on behalf of the middle market — small to mid-size manufacturing and service companies whose presence in international markets is limited and may even be characterized as sporadic, as compared to large multinational companies who undertake extensive globalization projects in order to serve numerous target markets simultaneously.[2] Thus, this article refers to middle market projects as translation and localization in the "real world." This of course does not imply that many of the issues addressed here are not also critical factors in large projects. The goal of this article is to explore quality from the perspective of each player in the "real world" subcontracting triangle, to delineate the unique quality requirements and constraints within that realm, and, ultimately, to propose practical paths to quality in translation projects.

This article uses the terms "translation service providers," "translation companies," "language service providers" and "agencies" interchangeably to denote organizations that market and provide coordinated translation and localization services to non-translation customers. The terms "translators," "freelance translators" and "vendors" refer to individual linguists, to whom translation companies subcontract translation, editing and proofreading services. On occasion, reference is made to translation companies and freelance translators collectively, which will be clear from context. When the words "clients" or "customers" are used, these signify the person or company that contracts with a translation company for services.

2. Obstacles to quality

Despite advances in technology (such as increased computing power, more stable operating systems and persistent, high-speed Internet access), improved access to qualified translators and greater awareness of localization issues, why is quality still a significant challenge to all the players in the subcontracting triangle? The answers are myriad.

2.1 Time and budget

Excellent translation should be the goal of any project. However, in the real world, excellence must serve two overbearing masters — time and money — which often conspire to transform excellence from an attainable goal into an impossible ideal.

Both translation companies and freelance translators are consistently subjected to time and price pressures. Prior to the advent of the Internet, the translation industry in the U.S. was a closed market. Buyers of translation had little choice of whom to turn to for services, and typically resorted to those listed in their local yellow pages. Over the past decade, however, due largely to the advent of the Internet, price control in the marketplace has passed out of the hands of translation companies. This is not unlike the impact Internet sales have had on the pharmaceutical industry, which has historically enjoyed nearly absolute price control in the U.S., due to the closed nature of that market.

The loss of price control has had a direct impact on quality, since quality-focused translation companies do not always have sufficient budgets to fund the proper processes and choose the appropriate translators for each project. Virtually limitless choice in the marketplace with regard to service providers — both translation companies and freelance translators — has caused consistent downward pressure on translation rates. Pricing differentials between the higher priced North American and Northern European markets and lower priced Asian and Latin American markets have only exacerbated the situation.[3]

The market-driven reality is that the demand for highly trained, specialized translators far exceeds the supply. The pool of qualified translators for virtually any specialized subject is small. Such translators do not work in house; they are independent (since they can earn far more in this fashion) and thousands of translation companies must compete with each other for their services. This dynamic mandates that high-quality, specialized translation cannot be cheap.

The ability to manage quality can also be compromised by limited time to effectively plan and carry out projects. Time, however, is just the corollary of money; it is subject to market forces as well. The root cause of time pressure, however, is not the high-paced information age work cycle, since technology has also made it possible for translators to work faster. The true cause of time pressure is as old as wax tablets: poor planning.

In the real world, most translation customers are service providers or manufacturers who are engaged in business-to-business (B2B) selling. B2B is inherently demanding due to the highly competitive marketplace. As a result of globalization these companies are not the sole source for a given product or service and must, therefore, respond faster than their competitors simply to win the business. This reactive position to customer demand does not provide for long-term planning or budgets for translation.

Indeed, insufficient planning for translation and localization is the result of a lack of clear, cohesive international marketing and sales strategy on the part of companies attempting international sales. Instead of proactively planning to enter specific national markets abroad, many companies instead target a larger region in the hope of augmenting domestic sales. Since they do not implement support for specific international locales, most companies must scramble to put that support into place when the need arises. If and when a sales opportunity presents itself, the client must rely excessively on the translation vendor, especially when it comes to meeting critical deadlines.

A corollary of this passive approach to international markets — even for smaller, nimbler companies — is indecision and tentativeness at pivotal moments in the translation/localization process. In most cases, once the need for translation has been identified there is typically a significant disconnect between the marketing/sales departments and the development/engineering arms within most companies. It is the sales department that is first aware of the opportunity to sell, by way of example, a top-line machine in Japan. For weeks, possibly even months, salespeople will pursue the sale with only a passing mention of the opportunity to the development managers, reasoning that there is no point concerning them with something that may not happen. Although this rationale makes sense, it also perpetuates passive behavior. It would be far better to fully engage the engineers and developers as soon as the sales opportunity materializes: even if the sale is not concluded, they will have gained valuable insight and experience with regard to localization and translation of that particular product.

The plea, therefore, from language service providers is simple: companies pursuing international business must prepare themselves and make translation/localization an automatic potential process for every product development effort. They should not allow it to be an afterthought, especially when estimating timelines and costs.

2.2 Lack of clear standards

The ability to achieve quality is significantly enhanced when there is a clear consensus as to *what* constitutes quality. Indeed, one of the difficulties faced by all translation service providers and buyers of language services is the subjective nature of quality. Unfortunately, the ISO 9001:2000 Quality Management Systems standard does not define a translation quality standard. The ISO standard simply requires that the service provider meet or exceed the customer's expectations of quality and/or ensure customer satisfaction. As highlighted in the previous section, the reactive position adopted by small to mid-sized companies with regard to international markets means that such companies are generally unable to specify the quality requirements for translation of documentation or localization of products. Translation services providers (both agencies and freelance translators alike) cannot rely on their clients to provide a quality standard.

2.3 Lack of communication

Another risk to successful projects stems from poor communication between the agency and the client. Often the agency may not have the proper materials, expertise or knowledge necessary to effectively complete the job. It is critical that the service provider take an active role in ascertaining the scope and the specifications of the project. All too often the agency thinks it knows what the project entails, simply to find out during the project that the effort needed to complete the project is far different than that which was originally anticipated.

Knowledge of the development process of the original product (document, Web site, software) is often lacking in dialog between the agency and the client. Usually the general processes are known and discussed, but the specific details and workarounds used by the client to optimize specialized software are not documented. For example, if a customer creates documentation that utilizes many images and screenshots, the language services provider should use the same processes and tools as the client to generate and capture localized screenshots. Knowledge gaps at this level of detail can cause significant delays in complex localization projects.

In addition to technical understanding and reverse engineering of processes, there must also be an efficient query and response system between the translation provider and the client. Clients should not interpret a lack of questions as a sign of competence. On the contrary, service providers whose translation teams do not ask questions do not perform as well. Likewise, agencies should be ready to answer customer queries about project status, processes and qualifications of the project team.

2.4 Garbage in, garbage out

One persistent "real-world" problem facing translators and the translation agency project managers who work with them is deficient quality in source materials. Translators, the agencies hiring them, and the clients who are providing the source materials will all logically agree that translation is more difficult if there are errors in the source materials. It is quite astonishing, therefore, how often poorly written, constructed and maintained materials comprise the source files of translation and localization projects. Agencies that have the good fortune to work with large clients that have in-house technical writing/communications departments may be accustomed to receiving well-authored and constructed documents. (Admittedly, large budgets and dedicated technical publications staff do not always guarantee this.) However, due to shrinking budgets in virtually all companies, full-time technical writers are becoming a thing of the past. In small to mid-sized manufacturing companies, technical communications are increasingly being assigned to engineering staff, which are ill-prepared for this added responsibility. Likewise, smaller software and Web development companies often do not have the resources and experience necessary to develop properly internationalized software. Internationalized software is designed to facilitate translation and localization for target markets. Poorly created documents and incompletely internationalized software problematize translation and localization efforts.

It is not uncommon for source documents to contain inconsistent terminology and usage, run-on sentences, ambiguous statements, unfinished sentences, incorrect factual information, poor orthography, and bad punctuation. Linguistic issues alone are often a significant hurdle to achieving high-quality translation. Add to that poorly applied styles in electronic files, under-utilized document automation and weak version control, and one has doomed the translation effort to failure.

The irony of having to confront such issues is that most clients are not even aware of the deficiencies in their source documents. The agency thus finds itself in a delicate situation, having to simultaneously quote and win projects from clients and criticize the poor quality of the source materials. Oftentimes, the client project coordinators are aware of the quality issues but are powerless or unmotivated to address them internally. The coordinators may be too low in the corporate hierarchy to exert any influence or they may fear possibly offending colleagues responsible for the production of flawed source materials. These issues impact quality on a daily basis. Of course, one would like to think that modern companies and their employees would be above such interdepartmental squabbles, but dysfunctional corporate behavior is more often the rule than the exception.

2.5 Technological barriers

The ability to accurately assess the technical requirements of a project, including the tools and methods used to create source products and documentation, is vital to the success of every project. The challenge to fully understanding technical requirements lies in the limited time available for project assessment. The knowledge transfer required to reliably localize a software product, for example, can be enormous: the agency must be familiar with the tool(s) used to author the software, and must know where translatable content is located within the application, what types of limitations may existing in regard to available space to fit text (i.e., string length limits), etc.

What about software development and authoring technology itself? If one falls into the trap of "best-case-scenario" project planning, one should be prepared to be bitten by the many bugs that persist in real-world technologies. For translators and localizers the list of technological threats is virtually endless. Since software developers and localizers use many of the same tools, the threats are common to both. Consider for example software that is used to convert Word and/or RTF documents to the Help systems that are part of most software programs. These Help authoring tools allow for on-the-fly creation of printable versions of the Help system. The only problem is that when one extracts the printable version, styles and formatting of the text are misapplied and become corrupted. If the same tool is used to create the original and the translations, both the developers and localizers will have to contend with this problem.

All too often the tools marketed for development of software and documents simply over-promise in the area of translation and localization. Many claim to be Unicode-compliant or to be able to handle double-byte languages (i.e., Chinese or Japanese). Such tools typically offer 90–95% reliability during actual projects. The devil lives in that last 5–10%. A good example is desktop publishing software that claims to support Unicode, and in fact does. One can import into this tool texts written in any number of different scripts (e.g., Hebrew, Cyrillic, Kanji, etc.). However, line wrapping and hyphenation of text that does not use Latin script is rarely supported, requiring that endless painstaking hours be spent manually breaking lines and sizing text in each document so that it fits into the original format and is consistent with the original. In this case, one must question whether or not the tool in question is the right one for the job. However, the answer is that the painstaking workaround required to make the tool work is probably still cheaper than recreating the document using another, perhaps more capable, formatting tool. Ultimately, the client must be aware of the technical limits of the software they choose in regard to translation and localization. By choosing software that is designed to support multiple languages, the client can avoid these technical hurdles from the outset.

Other culprits in the technical struggles of localizers are our own specialized tools. CAT (Computer-Assisted Translation) tools, or translation memory tools, are truly technical wonders. They allow translators to easily recall previously translated material within a specific project or for a particular client, hence the notion of "memory." The ability to recall and reuse pre-existing translation leads to significant cost savings, improved consistency and higher productivity. However, as is the case with any new technology, the productivity gains enabled by CAT tools are mitigated by the specialized knowledge and experience they require in order to derive optimal results.

The learning curve for independent translators is reasonable; most who adopt CAT tools can be back up to baseline productivity within five days of starting to use the software. Within a few weeks, they are often in the position to far surpass previous levels of productivity. However, much of the technical burden of implementing and maintaining the use of CAT tools falls on the translation agency. It is usually the translation agency's responsibility to prepare files for use with CAT tools. In most cases, this burden is not too onerous, since the tools are designed to work with most standard document, Web site and software file formats.

Nevertheless, these tools are not perfect. Translators often report that previously completed translations that should be in the translation memory simply are not there (or rather, are not identified as such and thus are ignored by the tool). Inexplicably, matches sometimes do not appear even though the segment in question has been translated before.

CAT tools' effectiveness also relies on the ability of the software to properly "segment" the source text. Segmentation entails splitting the text up at the sentence or paragraph level based on user-defined rules set within the tool, i.e., at each paragraph mark, full stop, hard return, colon, period, exclamation point, and/or question mark. Segmentation is a relatively straightforward process in most documents, but tagged files (e.g., HTML, XML, Xpress Tag, SGML, etc.) require greater care and setup time. Even with proper setup, segmentation problems are a common occurrence when translating tagged files, causing broken formatting in the translated documents (or a crash of the CAT tool itself), and adding additional time and costs not usually included in the project budget.

Another technical hurdle associated with specialized translation tools revolves around their development. Since the translation/localization industry is a relatively small market, the developers of the available tools have little influence on large software publishers (Microsoft, Adobe, Quark, etc.) in regard to standards and file convertibility for use with translation tools. If a new piece of software (for example, InDesign by Adobe) becomes popular and is adopted by document designers, then agencies will soon receive source materials in the new format, which may require the development of new processes or workarounds. Fortunately, although the CAT tool market may be small, the players in it have become more nimbly reactive and translation tool developers tend to add support for new file formats to their software fairly quickly. However, for translation companies and translators, these updates can never come quickly enough.

3. The translator's quality perspective

As we have seen, translators are not solely responsible for the quality of translations they create; myriad factors can affect their ability to effectively complete their task. But what do translators need to be positioned to perform optimally?

3.1 Time

The majority of independent translators equate sufficient time to complete a project with a high-quality outcome. Nevertheless, expectations are high among customers that translation should require less time than other steps in their development and manufacturing processes.[4] Educating clients about realistic timelines is a persistent issue for translators and agencies alike.

Proper project assessment by the agency is important to accurately determine the time required to complete a project. Likewise, detailed scheduling is vital to ensure that sufficient time is actually allotted to complete the project. Production standards for estimating translator productivity (i.e., "throughput," typically measured in words per day per translator) are relatively well established. For translators of English to/from Romance languages, turnaround of approximately 2,500 words per day is considered standard. For Asian languages, throughputs of 1,500–2,000 words per day per translator are considered standard.[5] Obviously, productivity can be impacted by the technical nature and/or quality of the source content.

Estimating turnaround times for large, complex projects is feasible using benchmark daily productivity metrics such as these. However, paralinguistic variables often make the planning of timelines more art than science. Assuming that the technical unknowns are minimal, "strategic" time must still be incorporated into the timeline to compensate for potential contingencies, be they technical or human. Surprisingly, project schedules rarely take human factors into account, since in the real world the expectation is that human factors should never impact a project! However the real-world project manager knows better and will silently plan contingencies and buffer time for contingencies such as illness and fatigue.

Agencies do have direct control over the timelines of their projects. However, in order to win more projects, many agencies will estimate project timelines too aggressively. This leads to the omission of key process steps from the timeline, or at least "skimming" time from these steps in an attempt to afford translators sufficient time. The "robbing-Peter-to-pay-Paul" approach to time management directly impacts translators, even if well intentioned, since typically time is being taken from upstream processes in order to afford sufficient time to translators and editors. From the translators' perspective, it is the upstream processes that are sorely lacking in many projects. Robbed time comes at the expense of critical steps such as glossary preparation, workflow planning, and, ironically, precise project scheduling. For example, proper terminology development, which is a critical set-up task that precedes translation, may require two full business days to complete for a given project. However, in order to give translators sufficient time, this may be condensed into one day. The result is a deficient project glossary, upon which translators must nonetheless rely to perform their work.

It should also be noted that timelines must be based upon the date of receipt of all source materials, and not upon the date of quote approval. If the delivery of source files to the translation company is delayed and the timeline is not adjusted accordingly, quality goals (and risk management strategy) will be severely undermined.

3.2 Clarity

Whereas allotting sufficient time to translators and editors is a prerequisite to meeting deadlines, a clear understanding of the subject matter is vital to the quality. All too often in the rush to produce (thanks to constant time constraints), translators are pressured to accept projects for which they may not be qualified or about which they know very little.

Translators who accept projects outside their area(s) of expertise risk not only the outcome of the project, but also their professional reputation and that of the agency that has contracted them. The reasons for translators to limit themselves to subject-matter domains in which they have extensive training or translation experience are as compelling as they are obvious: they understand the concepts they are translating and possess the requisite vocabulary to produce excellent translations.

Besides subject-matter knowledge, the translator must also have clarity of purpose in regard to the project. What is the purpose of the content? Who is the audience? What is the context of the words he or she is translating?

From the translators' perspective, a clear understanding of the context of everything they will translate is of primary importance. During translation of a homogenous document, such as a technical manual, most questions about context are easily answered. During more complex projects, such as software localization, the issue of context becomes more problematic. Since the localization of a software product involves translating text embedded in various parts of the software interface, deconstructing that context is required in order to access the information to be translated. This deconstructive process represents one of the greatest challenges for translators working today. Figure 2 shows an example of a software interface (in this case, a dialog box) as it appears in English. Figure 3 shows the source file with which the translation team must work in order to localize the software.

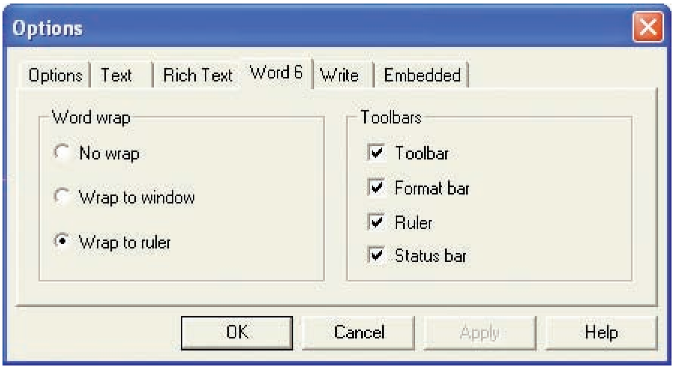

Figure 2. GUI (dialog box) of English software to be localized.

```
IDD_OPTIONS_WRAP DIALOG DISCARDABLE  0, 0, 280, 86
STYLE WS_CHILD | WS_VISIBLE | WS_DISABLED | WS_CAPTION
FONT 8, "MS Sans Serif"
BEGIN
    GROUPBOX        "Word wrap",IDC_BOX,7,7,128,72
    CONTROL         "&No wrap",IDC_WRAP_NONE,"Button",BS_AUTORADIOBUTTON |
                    WS_GROUP,13,21,81,10
    CONTROL         "&Wrap to window",IDC_WRAP_WINDOW,"Button",
                    BS_AUTORADIOBUTTON,13,39,81,10
    CONTROL         "Wr&ap to ruler",IDC_WRAP_RULER,"Button",
                    BS_AUTORADIOBUTTON,13,57,81,10
    GROUPBOX        "Toolbars",IDC_BOXT,144,7,128,72
    CONTROL         "&Toolbar",IDC_CHECK_TOOLBAR,"Button",BS_AUTOCHECKBOX |
                    WS_GROUP | WS_TABSTOP,153,21,68,10
    CONTROL         "&Format bar",IDC_CHECK_FORMATBAR,"Button",
                    BS_AUTOCHECKBOX | WS_TABSTOP,153,35,68,10
    CONTROL         "&Ruler",IDC_CHECK_RULERBAR,"Button",BS_AUTOCHECKBOX |
                    WS_TABSTOP,153,49,68,10
    CONTROL         "&Status bar",IDC_CHECK_STATUSBAR,"Button",
                    BS_AUTOCHECKBOX | WS_TABSTOP,153,63,68,10
END
```

Figure 3. Source file of same GUI in which localizers work.

Given the myriad types of software and technologies used today for written communications, it has been necessary and desirable to create tools to facilitate translation. Most CAT tools convert content from its original format into a neutral format that is easier for translators to work with, or else they filter the display of content to limit only material that should be translated. For example, TRADOS[6] has developed a specialized translation application called TagEditor™ specifically to facilitate translation of tagged file formats such as HTML and XML. Tools like TagEditor provide translators with easy access to translatable materials embedded into complex computer files, while simultaneously masking or protecting program code or other material that must not be modified. The tradeoff is that the translators may not readily be able to see the text in context, and are thus unable to ascertain whether the text appears in a headline, subheading, caption, etc. TagEditor compensates for this lack of context by offering a preview function as shown in Figures 4 and 5.

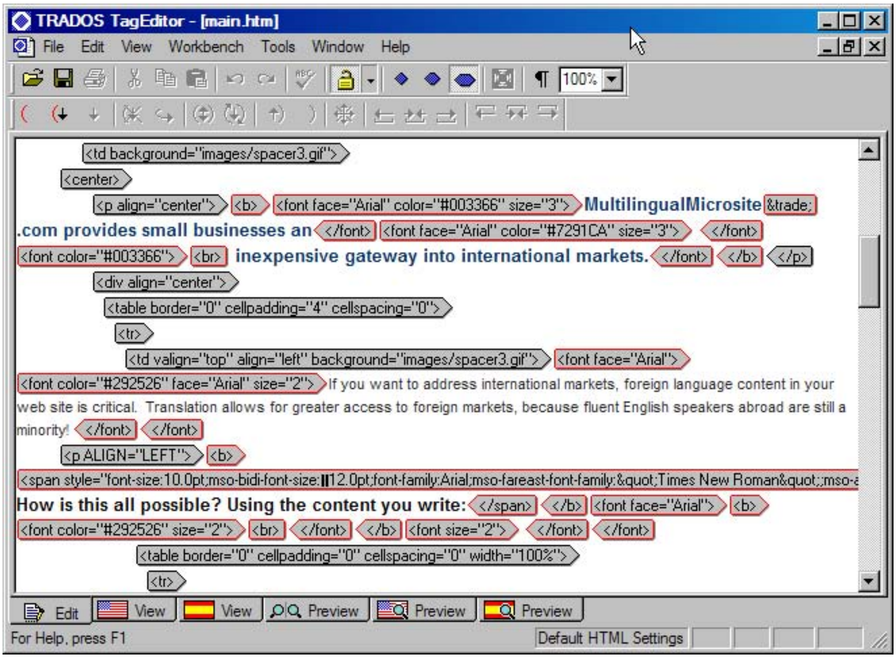

Figure 4. An HTML file in translation View mode within TagEditor.

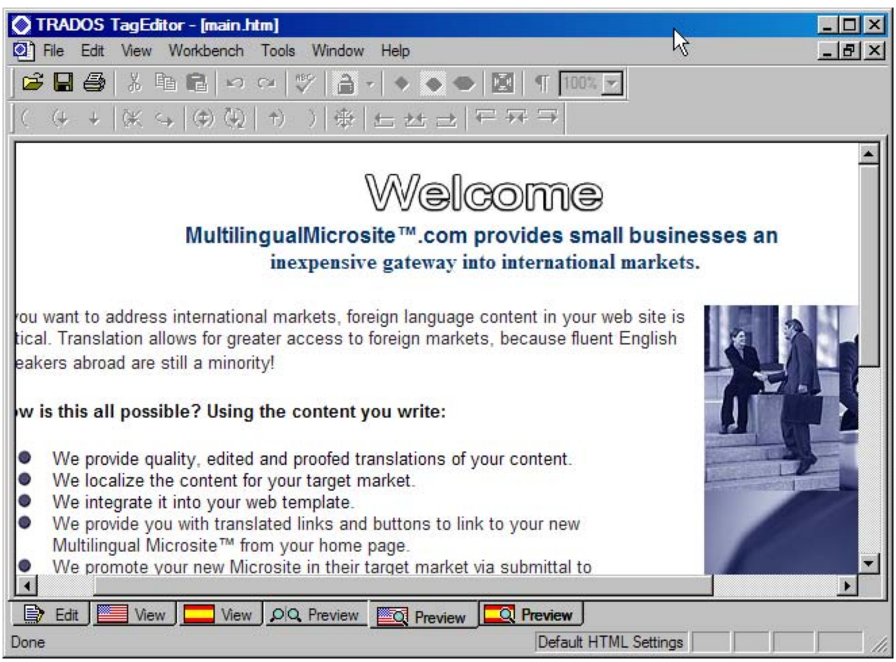

Figure 5. The same HTML file in Preview mode within TagEditor. Competitor products, such as SDLX, offer similar functionality.

3.3 Experience

Translators' ability to see and understand the context of the translated text is crucial to translation quality. In many cases, context is not as easy to grasp as in the example shown in Figures 4 and 5. General localization knowledge and project-specific experience are key factors in dealing successfully with complex localization issues.

Oftentimes, the content to be translated is for all intents and purposes disembodied due to the format in which the content is authored, stored, and/or maintained. A common example is the database content used to drive many software applications. Database tables store strings that the application will "call" when they are needed. Such strings may be displayed as labels in the software interface, values in menus, pull-down lists, dialog boxes or within pop-up messages that inform the user about the status of the application. For those unfamiliar with database tables, they are, as the name implies, information stored in tabular format.

If an application database is well designed, specific tables within it will hold content specific to different functions within the application. One table may contain only error message strings, another may contain only dialog box content, and so on. For an experienced localization translator, such a structure can aid in understanding context. However, this level of clarity is not enough to accurately localize a software application. There are numerous factors the localization translator must take into account in order to localize effectively. The various levels of project-specific knowledge are best described by the "forest-and-trees" analogy. When translating text strings that will appear in a software interface, translators naturally focus on the text immediately in their gaze. They may look at neighboring text to try to maintain a sense of context, but oftentimes their view becomes myopic as they become lost among the "trees" — thousands of text strings. To be effective localizers, translators must be able to see the "forest" — the whole product — and still maintain focus upon the "trees."

Forest — General application knowledge:

- Subject-matter expertise specific to the software
- Solid understanding of the software's purpose
- A mental map of the "topography" of the software (from where does specific content emanate, how are the various application data sources related to one another, etc.)
- Knowledge of style and usage preferred in the subject-matter area

Trees — Specific application localization experience:

- Knowledge of the localization history of the product

- Detailed knowledge of terminology specific to various components
- Experience dealing with technical constraints that impact translation (string length, string concatenation, the use of abbreviations, occurrence or prohibition of parallel translations, etc.)
- Knowledge of customer preferences in terms of terminology and usage

In order to produce quality work, translators must be given (and demand) sufficient time and information about the products and documents they are localizing. In addition, translators must be honest with themselves when considering a potential project: Are they specialized in the related subject matter? Do they have experience with the specific types of products and documents? Do they have the technical skills and tools required for the job? Above all else, they must maintain a cognitive balance between the holistic view of the product being localized and the detailed decisions they must make when working at the sentence level.

4. The agency's quality perspective

4.1 Translator competence

A continuing challenge to quality from the agency's point of view is a lack of translator competence. Translators are truly the foundation upon which any successful project relies. Given that the translators' performance impacts virtually every aspect of a translation or localization project, if the translation team is not up to the challenge, there is little the agency can do to ensure success.

The translation team's primary responsibility is to deliver well-translated content. The translation must be complete, accurate and stylistically appropriate for the target locale. Twenty years ago, such was the scope of the translation team's responsibilities. However today's technically demanding translation environments require far more from translators, who must be keenly aware of technical issues that impact the content they are handling. Put succinctly, from the agency's perspective, excellent translators cannot "merely" translate excellently; they must also be consummate users of technology in order to achieve optimal results.

4.2 Continuing quality

For quality-focused service providers, the ability to deliver quality projects consistently is of vital concern. Successful agencies rely on repeat customers in order to attain reasonable financial stability. Companies that do not consistently deliver quality from project to project will be unable to retain this critical customer base.

Continuing quality is further complicated in the middle market by the periodic, sporadic or opportunistic nature of many companies' translation needs. Tracking the needs, preferences and requirements of a client over time is more difficult when projects are undertaken on an irregular and/or infrequent basis. How, then, do real-world translation companies achieve "continuing quality?"

4.3 Team continuity

One of the best ways to sustain quality is to maintain the continuity of the translation team, whose members possess the tacit knowledge often required to consistently deliver quality. However, in the first-come, first-served world of independent subcontracting, translators' availability can never be taken for granted and it is thus nearly impossible to guarantee that the exact same team will work on each project.

The best approach that the translation company can take is to attempt to utilize the same team for the same clients from project to project but to occasionally bring new team members into the fold. To support the new and existing team members, the translation company should maintain the following resources:

- Client-specific style guide for each language supported
- Glossary or termbase for each client
- Up-to-date translation memories for each client
- List of requirements for deliverables
- Database of prior queries and respective answers
- Consistent Project or Account Manager

4.4 Knowledge management

To achieve true continuity, a robust Knowledge Management process must be put into place. Project Managers and the teams they manage (translators, editors, localization and desktop publishing specialists, terminologists, proofreaders and testers) must have the ability to quickly and easily record salient project information and share it across all the different language teams involved in the project, since the author's personal experience suggests that as much as 80% of the information contained by such a system may not be specific to a particular language. Current trends in knowledge management emphasize intricate, integrated technologies. In the real world, however, knowledge management merely means centralization and consistent use of termbases, translation memories, project logs, and query tracking databases.

4.5 Quality management systems

The Knowledge Management process should be part of a comprehensive Quality Management System that guides and regulates all processes that impact project quality and customer satisfaction. It is beyond the scope of this article to address Quality Management in detail, and many of the key factors that need to be controlled in order to achieve delivery of high-quality translation projects have already been addressed. In brief, a language services Quality Management System according to ISO 9001:2000 principles must:

- Document the quality philosophy of the organization
- Focus on continuous improvement
- Define a set of performance metrics
- Be focused on customer satisfaction
- Enable problem management and corrective action

When the translation of hundreds of thousands of words is involved, errors are statistically inevitable. Typos may sneak into documents; confusion about a specific technical term may arise; bugs in software are, unfortunately, a part of software development. This does not imply that such flaws are acceptable. It does mean, however, that there must be processes in place to minimize them and efficiently correct them.

A Quality Management System will benefit not only the agency, but also clients in terms of higher-quality service. It will also benefit vendors, since quality goals and expectations will be clearly defined and communicated by the agency.

5. The client's quality perspective

5.1 Ill-defined quality expectations

One of the major difficulties in translation and localization stems from the fact that many clients are newcomers to the process. Their companies may just be entering international markets, or they may have just committed seriously to a new market, which entails product and marketing localization. However, because of their translation and localization inexperience, many such companies often lack realistic quality expectations.

It is much easier to achieve a required level of quality if the quality specifications have been clearly formulated in advance. All too often in translation projects, translation teams (agency and translators alike) feel like they have been blindsided. A project will appear to be progressing nicely. Batches of files will be processed according to schedule; deliveries will roll in to the client who is encouraged by how well things are working. Then suddenly, inexplicably, difficulties arise. The client will report that there are terminology problems, formatting errors in documents are rampant, and there are bugs in the localized software.

There is obviously a disconnect somewhere in the process. The translation team's terminological "errors" may reflect more on the absence of clear target-language terminology guidelines than on deficient translation. Likewise, formatting "errors" may in fact be nothing more than the reproduction in the target text of inconsistencies that existed in the original documents. Bugs in the localized software may reflect existing bugs in the source file or arise from poor internationalization of the original software.

These issues must be discussed as part of a project's quality specifications. Terminology preferences can be accounted for prior to translation by preparing a detailed glossary, which the client should approve. Acknowledging formatting inconsistencies prior to translation helps identify problems, which can then be addressed by the localization team, thus improving the final product. In the same way, addressing technical and interface bugs in software prior to localization will avoid the need to debug two products (the original and localized versions).

Clients who lack clear, concise translation and localization quality goals present the following "symptoms":

- Unrealistic turnaround goals
- The belief that there is only one correct Translation for their content
- A blindness to the shortcomings of the source documents
- The belief that translation can compensate for or transcend all the flaws in the source material
- The expectation that translation vendors can immediately grasp complex content and contexts, master new concepts and reverse-engineer years of development work in a matter of weeks
- The expectation that the translated materials will be perfect for everyone, everywhere

An example will help illustrate this latter point. Often, clients will request "international" French or Spanish as the desired target language for a translation or localization project. By requesting the "international" variety of a language, they hope to address multiple target markets with just one version of the translated material. While this makes budgetary sense, it poses a conundrum for the agency and translators. While it is possible to translate for general audiences, locale-specific choices must nonetheless be made regarding the use of terminology, orthography and punctuation. In the case of French, for example, due to recent changes in rules governing punctuation in French-speaking Canada, it is less possible than ever to support both European and Canadian French markets with a single translation that will be universally acceptable. Agencies that continue to offer "international" translations for French and Spanish perpetuate the problem of client ignorance by confirming and reinforcing quality expectations that have no foundation in actual practice.

The scourge of unrealistic quality expectations can be blamed at least in part on translation companies themselves, which sometimes make unrealistic promises and claims. Too often, in order to win a contract, agencies will assure clients that they have teams of subject-matter specialists waiting in the wings ready to begin work when in reality, the number of true specialists in a specific subject matter area is so small that there is a risk that some team members may not have the depth of knowledge that is necessary for the best outcome if the agency is awarded the project. Agencies, although subject to the pressures of competitive bidding situations, must strive to maintain reasonable expectations with regard to capabilities, turnaround and cost.

It is also critical to commit shared expectations to writing. Documents that record agreed-upon standards and expectations enable the translation agency to measure its performance against a benchmark standard and serve to protect the interests of all parties. Such documents need not comprise a complete manual. Three items generally suffice: a style guide, a glossary and guidelines for client review and approval. Establishing and committing to writing even a minimal set of quality guidelines will have a profound effect on the quality expectations of clients, translators and agencies.

Client education is the best way for agencies to set clients' expectations and shape their standards regarding translation and localization quality. In addition, close collaboration between the agency and the client is the best way for both parties to maintain the appropriate quality perspective.

5.2 Client review

Most clients do not have the linguistic expertise necessary to assess quality first-hand. They are either at the complete mercy of the language services provider or are dependent upon in-country sales offices or their own customers to evaluate the quality of the translation. This inability to directly assess translation quality is often unsettling to clients and can cause apprehension towards the agency or perpetual skepticism about quality, even when exemplary service is being provided. Consequently, many clients contract with an external third party to conduct a vicarious "client" review on their behalf.

The client reviewer fills an important role in the translation process. He or she confirms that translations provided by vendors meet the client company's expectations of quality. The client review process is intended to be final approval of translated content prior to publishing. Ideally it is a collaborative process in which the client's reviewer, ostensibly a native speaker from the target locale with unique qualifications specific to the field in question, cooperates with the localization team to produce a product that is not only acceptable in the target market, but also reflects the linguistic preferences of the client company. Unfortunately, in many cases, the process is antagonistic rather than collaborative. The problem revolves around the central issue of quality expectations.

Frequently, the reviewer chosen by the client is unfamiliar with the translation process and may not even be a skilled communicator in either the source or target languages. In such situations, quality expectations remain unclear, since the direct client is not in a position to define them, and the designated reviewer, albeit a native speaker, has no basis for setting translation quality expectations. This scenario leads to a problematic situation in which the reviewer may make preferential changes to the translation. It is the client's, and by extension his designated reviewer's, prerogative to make changes to the translation. However, those changes do not reflect on the quality of the translation, since they are preference-based. Linguistic quality and criticism are often highly subjective, an issue that is usually overlooked by both clients and the reviewers. That is why the reviewer must undertake the task with an attitude of collaboration and avoid the temptation to impose his or her preferences upon the translation team.

Likewise, in-country review is fraught with potential entanglements, especially when the reviewer has a hidden agenda. Depending on the relationship between the home and remote offices, review may be seen as an unreasonable burden placed on the backs of overworked sales and marketing staff, a means to influence the marketing message being communicated by the home office, or as an opportunity for the remote office to show the home office how indispensable it is to the company. These political considerations have nothing to do with improving quality or defining appropriate quality expectations and make a dispassionate, collaborative review process nearly impossible.

Client/in-country review, if conducted properly, should have the following qualities:

- The reviewer must possess appropriate subject-matter and linguistic qualifications.
- Review should function as a quality control check and not as an editorial rewrite.
- The goal should be inspection, not line-by-line checking.
- Preferential changes should be avoided.
- Criticism must be given constructively and dispassionately.
- Errors must be reported clearly and concisely.

- Requested corrections must be unambiguous and actionable.

The reviewer's job is to confirm that the translation conforms to terminological and translation standards agreed upon with the client and report any/all errors. Errors are defined as:

- Overt mistranslations
- Use of incorrect terminology
- Failure to adhere to terminology included in client-approved glossaries, termbases, and/or translation memories
- Failure to adhere to established style guidelines
- Missing or incomplete translation
- Defects in orthography, typography or formatting

Review and approval of translated content by the client and/or a designated reviewer are important steps in the translation process. If conducted with an attitude of collaboration with an eye toward continuous improvement, the client will benefit from an efficiently developed, high-quality product, and the agency and translators will derive important feedback to improve performance.

5.3 Time-to-market pressures

Time-to-market pressure is exacerbated by the foreign sales dynamic of real-world client companies. Small to mid-sized specialized manufacturing companies do not have a predictable sales cycle in overseas markets. Since their products are specialized, demand for them is not widespread and sales are sporadic. Also, such companies tend to manufacture products needed by other manufacturers or businesses and require large capital expenditures. For these reasons, these companies must be able to move quickly when they uncover a sales opportunity in another country. Since such sales are opportunistic in nature, there is rarely a sales strategy in place to target a specific international market proactively. If this were the case, translation of marketing materials and product documentation would be undertaken well in advance of a sales negotiation.

The reality is that small specialized companies have no choice but to react: thanks to globalization and the Internet, they have both the blessing and the curse of being able to be found by anyone anywhere who may need their products or services. Given the competitive nature of global markets, small companies cannot afford to ignore a sales opportunity regardless of location. However, they lack the infrastructure necessary to drive proactive marketing into numerous overseas markets, hence the reactive time pressure conundrum that impacts all levels of the client's organization and the vendors who support them.

Time pressures can be minimized by all involved if the following steps are taken:

- The client company should pre-select a translation vendor who can support the client's current documentation/marketing processes for any and all regions most likely to generate sales.
- The moment an inquiry from a foreign buyer comes in, the language services provider should become engaged as would any other internal services provider (manufacturing engineers, technical publication departments, etc.).
- Client companies should be willing to make a small investment into developing a minimal set of marketing materials for target markets in which they believe they have the best sales prospects. Although this investment does carry some risk, it is small, and can save weeks of production time should a sale originate in one of the selected markets. Also, providing translated marketing and product information to prospective customers may actually induce a sale.
- Client company managers and executives should treat technical publications and translation/localization as marketing and/or development costs and recognize them as budgetary line items.

5.4 Translation costs vs. quality

The translation marketplace has become exceptionally competitive in the last five years, and as has been the case in most other industries, globalization has had a dramatic impact on pricing. Clients now have virtually an endless list of translation companies from which to choose, and are thus no longer limited to a local or regional pool of translation vendors. With broader choices come the price differentials that arise from diverse markets. Now it is possible to price shop translation from Berlin to Beijing. However, the quality may vary as much as the cuisine.

Everyone loves lower prices, but how low is too low? The answer is simple: the price must be right for the quality. If translation pricing is too low, it will be impossible to cover all the requisite costs of providing quality translation. Buying translation at rates far below the industry average likely means that something was missed when the quote was prepared, that the translation process may be lacking important steps such as editing or proofreading, or that unqualified translators are being used.

Overhead costs are a significant variable in the cost of translation, and of course there are places in the world where high-quality translation is cheaper because the cost of doing business in those places is lower. However, quality is viewed differently throughout the world. In certain markets, flaws in printed material may be common whether or not the material has been translated. Service providers in low-cost markets may not consider specific quality processes important or may simply conceive of processes differently due to their *own* quality expectations.

6. Paths to quality

Defining a set of quality expectations and processes to meet quality goals is far more achievable than writing an objective translation quality standard. Many of the ways quality can be achieved have been addressed above. The most important keys to quality are summarized below.

6.1 Quality source materials

All translation efforts benefit when source materials are of high quality. Clear and concise source documents will enable equally clear and concise translations. The source text must also make consistent use of terminology that is appropriate to the topic and the audience. For most documents, a neutral style that avoids jargon and uses imperative mood and active voice is preferred. The source materials must be well constructed technically. In documents, for example, the correct application of styles is important so that the translated version can easily replicate the layout of the source, thereby saving time and reducing costs. If images are used, they should be well organized and clearly labeled in separate directories in the digital source files. All images embedded in documents and elsewhere should be provided in the form of editable, layered artwork files (e.g., *.PSD [Adobe Photoshop files]).

6.2 Mastery of the basics

With reliable source materials in hand, clear instructions from the client, a reasonable budget and sufficient time, it is up to the language services provider and the freelancer(s) to simply do the job. Often service providers over-complicate translation and localization processes. For example, sophisticated software may be used to mine terminology from a relatively small set of documents, whereas a manual approach involving a linguist who reviews and compiles a glossary himself may well be more time- and cost-effective (since terminology extraction tools typically present substantial numbers of spurious term candidates). Too often, technology acts as a crutch for weak production processes or poorly skilled linguists.

For most translation service providers, greater focus on basic best-practice processes would benefit both them and their clients more than technology. The most important processes are those that govern day-to-day human transactions in the translation and localization workflow. The best processes are simple, easy to follow, and ensure universal compliance among translation and localization workers. For example, a basic process that defines how files are to be managed and stored at all steps of a project will minimize confusion and help ensure that each team member has the right files at the right time.

Translators must place more emphasis on basic, critical tasks such as research. It is astounding how often agencies are peppered with questions by translators who can easily find answers to basic content questions by doing an Internet search. On the agency side, localization specialists must spend more time on project analysis to better understand the structure of software and Web sites to be localized in order to provide linguistics with the technical support they need to perform optimal work.

6.3 Consensus on quality

Consensus can be reached in regard to translation quality, as the DIN 2345 standard demonstrates. The DIN 2345 Translation Quality Standard specifically defines how a translation project must be executed, which types of information are required and what the qualifications of the participating workers must be (Sturz 1998).

DIN 2345, which can be seen as a practical model for the real world, specifies the following translation qualities:

- Completeness
- Terminological consistency
- Correct grammar and appropriate style
- Adherence to an agreed-upon style guide

DIN 2345 is less a quality standard than a set of expectations with regard to quality. Fulfillment of these expectations will certainly have a positive impact on quality. Completeness, grammatical and stylistic quality are relatively easy to achieve if the translation team is attentive and is translating into its native language. Terminological and stylistic consistency are more problematic in the real world due to the time and budget constraints to which most projects are subjected. The time pressure is often such that there is no time for proper glossary development before commencing translation. Target-language style guides are even more problematic, since most clients have no formal stylistic expectations in languages other than their own (and may not have even codified stylistic requirements for their source-language materials). A proactive agency will develop general stylistic standards for all the languages in which it works, but the agency's standards may not satisfy the client's in-country reviewers or customers (assuming these vicarious reviewers are even identified in advance). And there is rarely time to ascertain what those specific stylistic expectations may be, let alone document and communicate them to the project team.

The other critical facet of most translation and localization projects is effective integration of the translation into the final product. Integration encompasses both insertion of translated content back into the layout of the original documents as well as into software interfaces and Web sites. Translation quality may be exceptional, but if the translation cannot be effectively and reliably integrated into the final product then one cannot speak of an overall quality outcome for a project. Integration is more process- rather than performance-driven as compared to translation. As such, it requires a more robust quality process management framework.

6.4 Quality management systems

Establishing a formalized Quality Management System (QMS) will give translation companies a framework within which to build, document and manage the basic and complex processes that ensure high-quality services. A QMS must be more than a document repository that contains grand proclamations of the ways in which quality will be achieved. More importantly, the QMS must illustrate the ways in which the company plans to implement or put into practice its core philosophies regarding customer service. Its employees' and vendors' adherence to the QSM should follow the spirit of what the quality system strives to achieve, rather than devolve into rote adherence to an audit trail. In other words, the QMS structure exists to reinforce proper behaviors and interactions that yield continuously improving quality.

6.5 Quality translators

Any agency struggling to deliver great quality to its clients and looking for areas in which improvements can be made should begin by assessing vendor performance. It is incumbent upon agencies to set the standards and specifications to which their vendors must adhere.

Agencies must carefully scrutinize the qualifications of new and existing translator vendors. Proper vendor assessment includes work experience, educational focus, professional focus and technical capabilities. Today, excellent translators can no longer be identified based solely on linguistic ability. They must be excellent users of translation memory and terminology management tools, consummate Internet researchers, accomplished users of standard business software (e.g., MS Office, WinZip, etc.) and have minimally a conceptual understanding of desktop publishing and software development. Without strong technological skills, it matters little if their translations are excellent, since it is impossible to deliver excellent quality on time and on budget without efficient use of what have become the standard tools of the trade.

6.6 Terminology management

One of the most frequently neglected paths to quality is terminology management. For translators, terminology management is the most important type of knowledge management. Nevertheless, it is ironically one of the first steps to be omitted or truncated when project scheduling faces tight deadlines. Agencies may often assume that translators will manage their own terminology while working. Minimally, good translators will maintain simple bilingual term lists, but such lists are of limited benefit, since they are not accessible and cannot be shared by the project team as a whole. It is the agencies' responsibility to implement terminology management for each project and then guarantee continuity of each termbase from project to project. It is the translators' responsibility to expect a well-maintained termbase for nearly every project and then use and maintain termbases consistently during projects. Clients must also expect terminology development. They must hold translation agencies accountable for creating and maintaining glossaries and be willing to pay for terminology development.

6.7 Translation memory management

Much like terminology management, translation memory management plays a vital role in helping to ensure consistent quality. Translation memories typically overshadow glossaries, since they have a measurable impact on project costs and productivity. As is the case with terminology management, translation memory management is the primary responsibility of the agency. However, translators play an important role in successful TM management. They must be cautious not to blindly trust translation memories provided by agencies and should consistently confirm TM validity. Clients must educate themselves about translation memory tools, weigh the pros and cons of the various available tools, and hold agencies accountable for proper setup and maintenance of translation memories. They must also recognize that while translation memories deliver cost benefits, they also entail implementation and maintenance costs.

7. Conclusion

Hard choices must often be made in regard to issues of quality. What will budgets and time allow? Are clients aware enough to even value critical processes that enable high-quality translation? The answers are that translation companies and the translators with whom they associate must forever balance the competing imperatives of cost, time and quality. In order to do so they must have a clear understanding of the quality requirements of each project and every client. They must also adhere to a set of processes that govern all of their translation transactions, and, lastly, commit to management of knowledge, terminology and translation memory resources. Above all, they must continue to educate the client, for it is in the middle market where the power and potential of global markets conflicts head-on with the naiveté and blind optimism of small and mid-sized companies who must capitalize on international opportunities in order to thrive.

Notes

1. Based on an Internet market research study conducted by my company, Advanced Language Translation Inc., in March 2004.

2. This is an empirical assertion based on comments by and overall demographics of member companies of the Association of Language Companies (www.alcus.org).

3. Some languages and markets have been more resistant to downward pricing pressures than others. For example, per-word Japanese translation rates have generally not dropped as dramatically as Chinese rates.

4. It is not completely clear why this expectation exists among many clients; however, the advent of automated (machine) translation in the marketplace may play a role. Many clients without prior experience with translation immediately assume that automatic translation is used by most translation companies and is therefore fast.

5. These *de facto* standards are so well accepted that translators' CVs typically include a "words-per-day" section.

6. TRADOS Inc. merged with SDL International in July, 2005.

Reference

Sturz, W. 1998. "DIN 2345 und die Auswirkkungen auf die Übersetzungsbranche." *Doculine-News*, December. http://www.doculine.com/news/1998/12_98/DIN2345.htm

Putting the cart behind the horse

Rethinking localization quality management

Keiran J. Dunne

"Intellectuals solve problems; geniuses prevent them." — Albert Einstein

Introduction

In 1964, during a U.S. Supreme Court case involving the appeal of an obscenity conviction, Associate Justice Potter Stewart noted that he was incapable of strictly defining pornography and added, "but I know it when I see it" (Supreme Court 1964).[1] Much the same could be said about quality. For example, if we ask a number of people to indicate whether a given automobile is a quality product and why, we will undoubtedly receive multiple replies ranging from reliability and fuel economy to the driving experience, aesthetics, quality of dealer service and even the relative ease of obtaining replacement wiper blades (Lommel 2004).

So, how can we define quality? In the realm of manufacturing, quality can be objectively measured by performing tests, assessing tolerances, failure rates and/or purity of raw materials, and by gauging the repeatability of processes and the reliability of products. However, defining quality is much less straightforward when the product in question does not lend itself to objective measurement. For instance, how might we define the quality of a novel? We could focus on the formal characteristics of the text, such as the number of spelling mistakes and grammatical errors or the relative syntactic complexity, but most readers would be far more interested in the originality of the work, the nature and scope of the plot, the richness of the character development, and so forth. The problem is that such criteria are highly subjective compared to the objective, quantitative benchmarks used in manufacturing.

Software represents a synthesis of these two extremes. Because the functional aspects of a program can be objectively assessed (via testing routines, for example), people often perceive software development and localization as processes that are akin to manufacturing, while forgetting about the linguistic aspects of the program and the often subjective nature of linguistic quality definitions. In any event, if the program is to be localized, any definition of quality will have to expand to take account of the linguistic, cultural, legal, technical and regulatory differences of the various target locales.

The inherent difficulty of merely *defining* quality in the language industry raises questions as to how and to what extent quality can be *managed* in localization projects. Currently, the most widely adopted approach is that of *ISO 9001:2000, Quality Management Systems*. Using the ISO 9001:2000 standard as a frame of reference, this article will discuss the current state of localization quality management in terms of theory and practice, and will explore some of the quality management issues that can (and often do) arise due to the fundamental tensions between ISO 9001 principles and the localization project model that dominates in the current marketplace.[2]

The evolution of quality management

In order to better understand the foundation and rationale of current ISO quality principles, it is helpful to briefly examine the evolution of quality management, which has undergone a profound shift over the past century. Prior to World War II, each generation of workers trained the following generation in quality production methods, and quality product depended on the inherited transmission of knowledge from journeymen to apprentices. However, during the 1940s, due in great part to increased quality requirements on the part of military customers, many manufacturing companies began to rationalize quality management with an eye to decreasing liability and the relative frequency of discrepant product. Specifically, this involved a commitment to Quality Control (QC), namely a defined set of procedures designed to confirm that the product or process conforms to the requirements set forth by the project stakeholders, and relying on end-item inspection (whether systematic or random) to identify and quantify the frequency of non-conforming product.

Nevertheless, inspection proved to be very time- and labor-intensive. Beginning in the 1970s, management thus sought to shift the burden of work (and liability) of ensuring quality away from inspection and correction, back toward the production process and producers themselves (see Figure 1). In so doing, manufacturing companies shifted their focus from Quality Control to Quality Assurance (QA) by developing and implementing risk management strategies designed to proactively identify and mitigate future negative contingencies in order to ensure the quality of the final product. QA is fundamentally different from QC in that it is a process- or management-oriented approach that ensures quality via

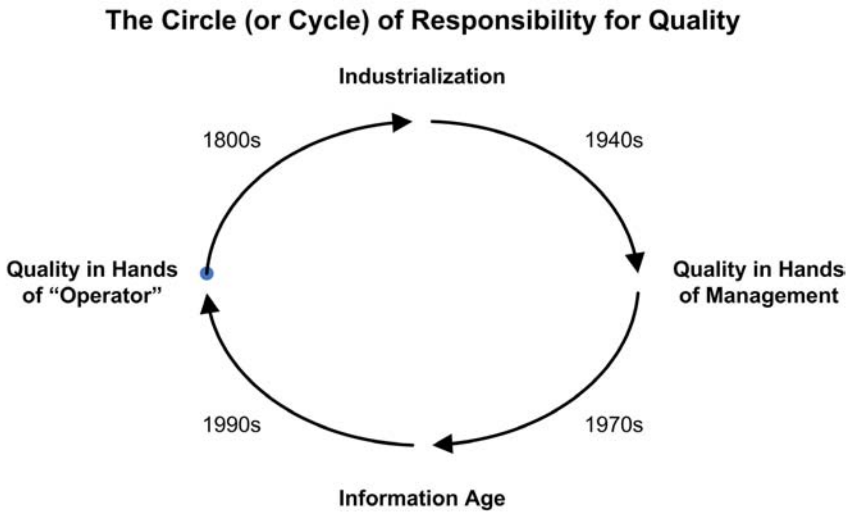

Figure 1. The circle (or cycle) of responsibility for quality, (Smith 2004:3).

the prevention of non-conformance. Unlike QC, QA focuses less on *what* is made than *how* it is made and implies that conformance with quality standards has been assured through inspection, either by internal or external clients.

The QC-to-QA shift from correction to prevention and from product to process is mirrored by a shift in the definition of quality itself. The predecessor of the ISO 9001:2000 standard, *ISO 8402 Quality Management and Quality Assurance* defines quality as "the totality of characteristics of an entity (or process) that bear on its ability to satisfy stated or implied needs" (PMI 2000: 96). This definition of quality requires that we identify the characteristics of the localized deliverable (or of the processes by which the deliverable is produced) that have an impact on its ability to fulfill its function. ISO 9001:2000, on the other hand, broadens this functional definition by framing quality in terms of the "ability to consistently provide product that meets customer and applicable regulatory requirements" (1), thus emphasizing the repeatability of the process.

The process-based approach to quality management

Quality does not simply happen; rather, it is the result of project planning, policies and processes designed to achieve a quality product. A quality management plan for a project should describe how the project management team will implement its quality policies. "In ISO 9000 terminology, it should describe the *project quality system*: 'the organizational structure, responsibilities, procedures, processes, and resources needed to implement quality management' " (PMI 2000: 99).

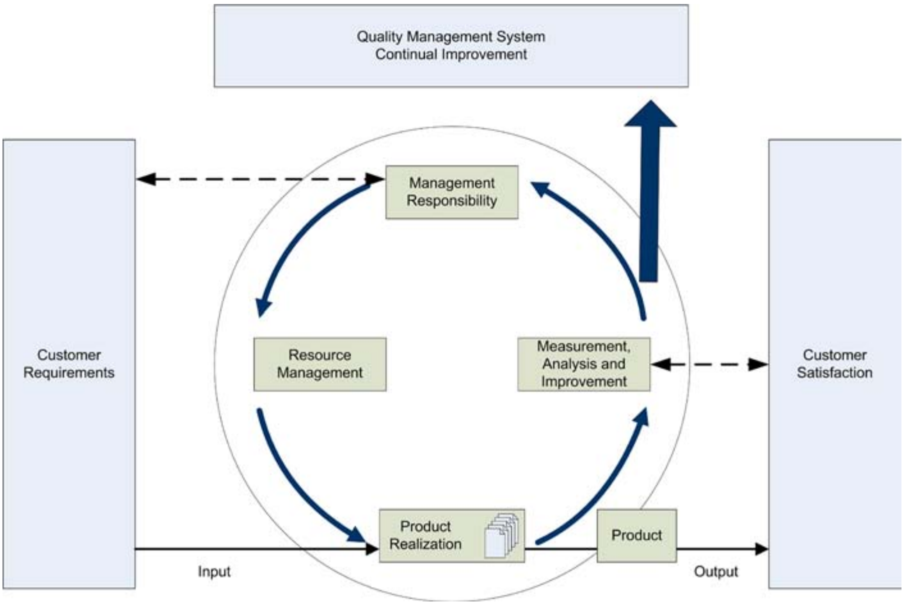

Figure 2. A process-based approach to quality management. The solid arrows represent value-adding activities, whereas the dotted arrows represent the flow of information.

As seen through the ISO 9001 prism, achieving quality is a matter of specifying and meeting client requirements (see Figure 2). Prior to beginning a project the client and vendor consult in order to ensure the formulation and communication of the client's quality requirements, which form the standard by which the quality of the product is evaluated. The vendor then develops and implements the quality project and product plans needed to meet the customer's requirements. Collectively, this set of quality project and product plans constitutes the vendor's quality assurance processes. During the realization of the product, measurement and analysis of the output are performed at each step to ensure compliance with the vendor's QA processes and standards. These verifications, which comprise the vendor's QC procedures, ensure the application and conformance of QA strategies. Upon completion and delivery of the product, measurement and analysis are conducted with respect to the client's quality requirements specification, and feedback is solicited from the client in order to identify areas for improvement. In sum, the ISO 9001 standard "promotes the adoption of a process approach when developing, implementing and improving the effectiveness of a quality management system, to enhance customer satisfaction by meeting customer requirements" (2000: v).

ISO 9001: Quality management for/in manufacturing

The ISO quality management standards were developed by and for manufacturing, based on the principles of Total Quality Management (TQM), whereby quality is evaluated at each step of the process to identify nonconforming or defective product, per client requirements, in order to increase overall quality and efficiency while reducing the need to subject every end item to time- and labor-intensive inspection (see Figure 3).

In the realm of manufacturing, the quality management model shown in Figure 3 presupposes that:

- the organization has contracted with a client to produce some sort of product, from scratch ("scratch" being loosely defined here — production could involve transformation of raw materials, integration of components or subassemblies, etc.);
- the client has provided the organization with a detailed requirements specification;
- the project will begin only after global project requirements have been formulated;
- the organization will procure its parts and/or raw materials from third-party vendors based on the quality and price of their offerings;

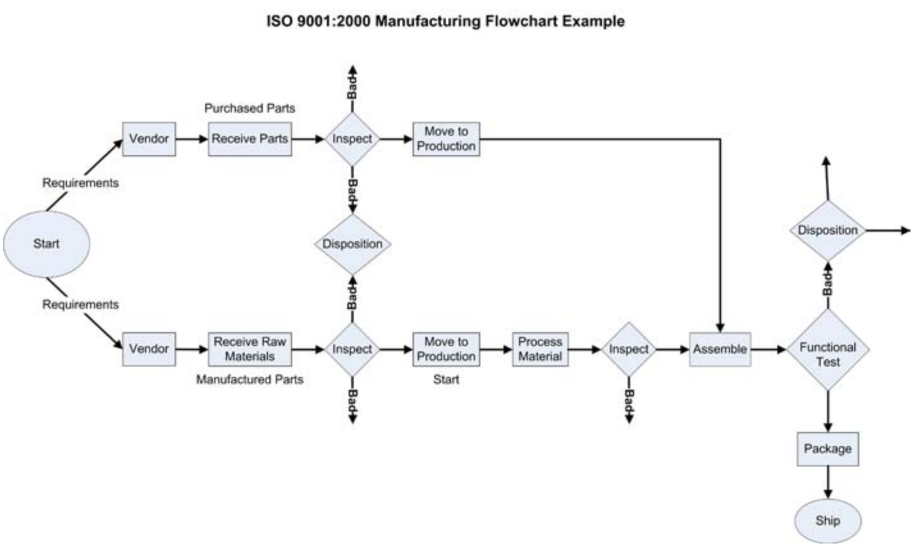

Figure 3. ISO 9001:2000 manufacturing project flowchart example, courtesy of The Elsmar Cove.

- the quality of the raw materials, components or subassemblies used in production can be measured quantitatively and objectively, in terms of tolerances, failure or defect rates, and so forth;
- the manufacturer has direct control over the quality of the input provided by the vendor (via communicated requirements); if the quality of vendor materials is deemed inadequate, the manufacturer will declare them unfit for use in the project during receipt and inspection of raw materials or purchased parts;
- the organization is provided with a blueprint for production or will create one prior to engaging in production;
- consistent adherence to the processes documented in the quality management system will ensure the reproducibility and repeatability of product quality, measured quantitatively and objectively in terms of tolerances, failure or defect rates, and so forth.

ISO 9001 for the language industry: An anachronism?

In a typical outsourced localization project, by way of comparison, the organization does not contract with a client to *produce* a product, but rather to *adapt* an *existing* product (see Figure 4). In this case, the goal of the project is not so much to create a client-defined deliverable but to adapt a client-produced receivable. The client–vendor relationship thus differs in fundamental ways in localization compared to manufacturing. Localization is more akin to custom retrofitting than to manufacturing from scratch. Nevertheless, clients tend to apply the same standard and model shown in Figure 3 to the localization and translation processes shown in Figure 4, assuming that the vendor can (and ideally, will) ensure all aspects of project and product quality from project kickoff to delivery. This represents a serious breakdown in the application of customer-focused quality management principles, since the "raw materials" and "parts" are supplied by the *client*, not by a third-party vendor, and the organization has little or no control over the quality of the source materials, nor is there at present any standardized means to quantitatively and objectively evaluate the quality thereof.[3] Rejection of "deficient raw materials" in a localization project is not an option if the vendor wants the business! And yet according to any objective definition of quality, the use of flawed or defective source materials will seriously undermine the quality of the finished product.

Compounding these problems is the fact that comprehensive quality requirements or specifications are almost never provided in localization projects. In fact, paradoxically, clients often *cannot* provide all the necessary standards, requirements or specifications for the simple reason that they are unfamiliar with the languages, cultures, conventions, and legal requirements of the target locale(s).

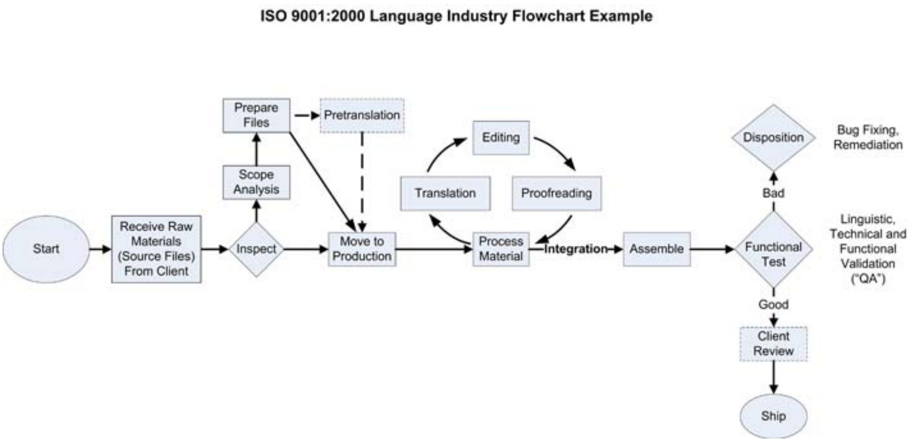

Figure 4. ISO 9001:2000 principles as they are typically applied in the language industry today (TRADOS 2004: 3; Scribe Consulting 2003).

Indeed, this lack of knowledge and expertise is a major factor in many clients' decision to outsource localization in the first place (Shreve 2000: 226). In addition, the localization needs of many clients are driven not by proactive international business development, but rather by opportunistic sales. In the event that a sale materializes, such clients outsource localization into the required language(s) in order to avoid the headaches of doing (or managing) the work themselves.

Regardless of the root cause, the two-way flow of information and communication about requirements shown in Figure 2 is often deficient in localization projects. In the absence of *identified* needs (requirements), quality is defined in terms of *unidentified* needs (expectations). In such cases, the responsibility for defining quality falls upon the vendor by default. Framing a project's quality management in terms of unidentified needs is fraught with risk, since the client's tacit quality expectations may ultimately prove quite different from those of the vendor.

Comparing Figures 3 and 4, we see that the start of an outsourced localization project typically corresponds to the receipt of a completed product, followed by reverse engineering and the subsequent production of a localized version based on the study, analysis and emulation of the model. Comprehensive specifications are generally not provided; they are partially *derived* from reverse engineering of product, partially communicated by the client and partially determined by the vendor or subcontractors according to their best judgment. This approach is at odds with the customer-driven ISO 9000 quality model shown in Figure 2, and may introduce a considerable amount of risk into the project. It is far riskier to derive implicit requirements from a finished product than to apply explicitly articulated ones. In the realm of manufacturing, an analogy of an outsourced localization project with unspecified quality requirements would be a project in which a client presents a vendor with a finished heat-transfer machine and requires production of an equivalent, without specifications.

So, why do we insist on drawing a clear distinction between ISO standards as they are applied in manufacturing and in the language industry? The answer is that this distinction raises a number of fundamental issues that force us to question notions of quality, quality processes and the very translation process itself. Moreover, the impact of the failure to draw this distinction is arguably greatest in the realm of localization, since localization is the language industry sector in which process automation is most prevalent and most critical, due to the large volumes of material, large number of target versions that may need to be produced for simultaneous shipment, large number and wide range of file formats, as well as cost and time-to-market pressures. It is in the localization sector that repeatability and reproducibility of process (or lack thereof) most directly translates into large savings (or large costs). Localization is also arguably the sector that can be considered the closest to manufacturing due to the highly pragmatic nature of the typical source text, as opposed to those in other types of translation.

Software and other localization-related texts are pragmatic, functional and informative in nature. Therefore, the content of a software program should — in theory — be denotative and relatively transparent. Unfortunately, there is often a substantial gap between the theoretical attributes of the source text and the actual characteristics of the materials to be localized. In localization, much of this gap can be attributed to lack of context.

In a "classic" software localization project, in which GUI resources are stored in external files such as satellite DLLs, the relationship between strings and the objects with which they are associated is relatively clear when working in mainstream localization tools such as Alchemy Catalyst, Passolo, or SDLinsight.[4] Such tools enable users to toggle between a string view or editing mode, in which strings are displayed in a tabular fashion (see Figure 5, left-hand image), and a WYSIWYG mode (i.e., "what you see is what you get"), in which strings are shown in the context of the menu or dialog where they display at runtime (see Figure 5, right-hand image).

However, over the past few years, reflecting the shift away from desktop applications to Web-based applications, and in an effort to streamline code and to facilitate development as well as the reuse of content, many software publishers have begun to move away from the classic model of software authoring, in which GUI resources are stored in the executable or in satellite DLLs, in favor of more flexible XML-based approaches. XML allows for the absolute separation of form and content, and also is extremely flexible in that developers can create their own tags, schemas and document type definitions. XML offers many advantages to developers but raises serious issues for translators due to the extreme lack of context

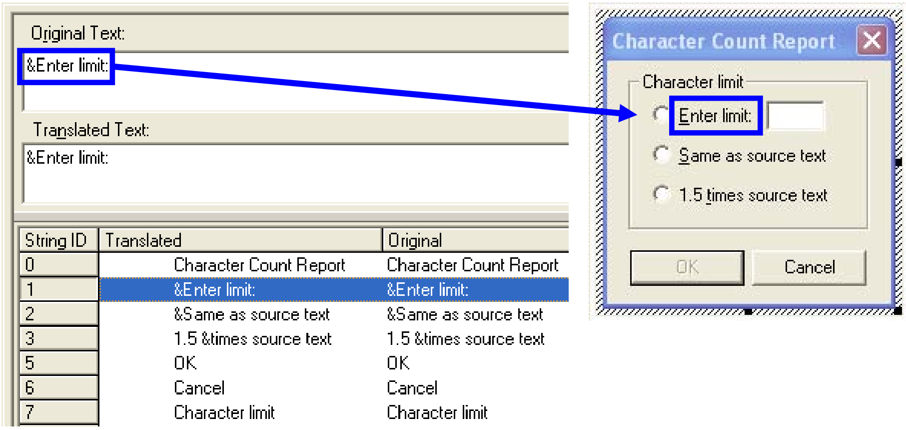

Figure 5. Display of ForeignDesk strings in editing mode (left-hand image) and WYSIWYG mode (right-hand image).

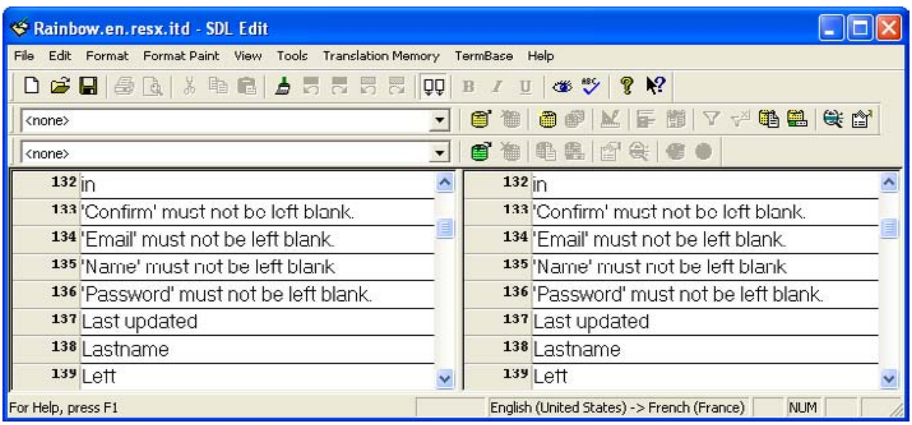

Figure 6. Translation of application GUI resources stored in *.resx format (Microsoft® .NET XML) using SDLX®.

they must overcome while working with files in this format. Because it does not lend itself to WYSIWYG editing, XML obfuscates the relationship between strings and objects, and this problem is amplified when translators work in CAT tools that hide all tag information and display strings for translation without any context whatsoever (see Figure 6).

When working in such files, translators may be unable to determine the function and context of a given string (menu, dialog, caption, etc.), and thus the appropriate target-language equivalent. Figure 6 illustrates the lack-of-context conundrum nicely. For instance, the fact that string 132 is not capitalized and consists of an isolated preposition suggests that it is concatenated. Translation of this string into German would require identification of the gender, number and case of the concatenated object string(s) in order to determine whether or not there will be contraction (the preposition will be "*ins*" if the object is accusative neuter, but will be "*im*" if the object is dative masculine or neuter). Accounting for all the possible target-language equivalents would require the addition of as many new strings to the code as there are additional possible combinations in the given target language. Conversely, it is also necessary to identify the verb with which the preposition "in" collocates, because the target-language preposition may not be "in" at all. Similarly, translation of string 137 into French would require identification of the noun modified by "updated," since nouns and adjectives must agree in gender and number in French. Otherwise, recasting would be required to avoid this problem, for instance by translating "Last updated" as "Last update." In this way, the French equivalent "Mis(e)(s) à jour le" would become "Dernière mise à jour :" Translation of string 138 raises the question as to whether "Lastname" should be translated, since word combinations are often non-translatables (Esselink 2000: 69), or whether this is simply an example of "developerspeak" in the source text (i.e., "Last name" incorrectly written as "Lastname"). Last but certainly not least, translating string 139 into Russian would require knowing whether it is an independent adjective (*gauche* in French or *links* in German) or a pre-positioned modifier (i.e., "remaining"), since in the abstract there are twenty-odd different potential translations of "left" into Russian. The text as displayed in Figure 6 does not enable a determination to be made with regard to most of these issues.

The source code is not often much help either in such instances. Unlike source code of traditional resource files that explicitly identifies menu items, dialog box titles, button labels and other GUI items as such, there is no guarantee that XML tags will elucidate the GUI location/function of a string unless they have been written with this goal in mind. The source code of the file shown in Figure 6 clearly was not, since the data name attributes of our problematic strings 132, 137, and 138 are virtually identical to the strings themselves, capitalization and underscores notwithstanding (see Figure 7).

In the absence of contextual information or data name attributes that shed light on the meaning of the string, the localizer has little choice but to submit a query. If these problems are widespread across a project, the number of queries can quickly spiral out of control and become a logistical nightmare for localizers, the localization vendor project manager, client-side project manager and developers.

The use of XML thus frames the problem of the forest and the trees in stark contrast and presents a rich paradox. The way in which mainstream CAT tools handle XML forces the translator to focus on the trees (individual strings in the CAT tool), whereas accurate translation requires that the translator see, understand and account for the trees, the forest (the overall structure and architecture

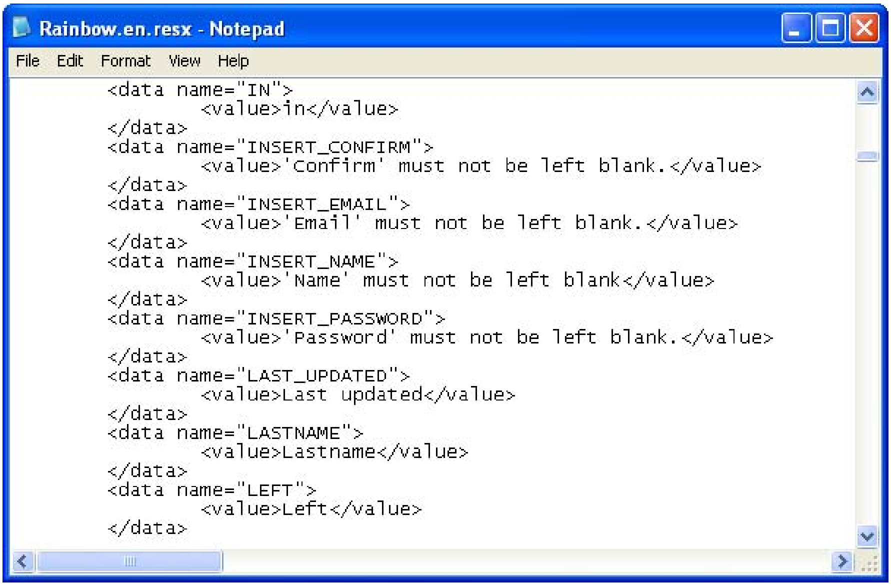

Figure 7. Source code corresponding to the portion of the GUI resource file shown in Figure 6.

of the entire application) and the relationship between them. Often, the effort required to simply determine the part of speech, function, and precise context of usage in order to ensure a correct translation is such that little attention is (or can) be paid to stylistic concerns. In such cases, textual localization is transformed from an attainable goal to an unrealistic ideal.

Although Figure 6 represents a somewhat extreme example for purposes of illustration, it underscores the potential problems raised by the separation of content and context in XML and clearly shows the impact that the file architecture can have on the process of GUI translation. Although XML may be a godsend for developers, the complete absence of context when viewing XML in certain CAT tools undermines the efficiency of translation, and risks undermining the accuracy as well.

This is not to say that translation problems are caused only by the use of non-internationalized XML — on the contrary, translation efforts can be (and often are) seriously undermined by the following source file problems:

- File structure and content that does not reflect the logical structure of the component(s), but rather the chronological history of development work or some portion thereof
- Help, user assistance and documentation that do not reflect the actual content of the user interface

- Inconsistent source terminology (the use of multiple terms to refer to a given concept)
- Overuse of jargon
- Concatenation
- HTML embedded in XML
- Ambiguity as to part of speech in single-word strings such as "archive," "override," "update," and others that function as both nouns and verbs in English
- Structural calques (using nouns as verbs and vice versa)
- Non-standard usage of standard terminology
- Non-grammatical usage
- Inappropriate use of metaphorical language
- Inconsistent stylistic rules regarding capitalization and punctuation of GUI terms
- The use of competing orthographic rules and standards during authoring of source files
- Inconsistent application of styles and formatting
- Typos, grammatical errors, and errors of fact

This list is by no means exhaustive but reflects the most commonly encountered source-text problems (based on the author's personal experience).

Having examined some of the more common problems encountered during nuts-and-bolts localization work, let us contextualize them by turning our attention back to the larger context of the client–vendor subcontracting relationship. It will once again be useful to draw a distinction between the client–vendor relationship in the realm of manufacturing (the paradigm from which ISO quality management principles are derived) and this same relationship in the language industry.

In a typical manufacturing project, the perspectives of the client and the vendor are fairly linear and emphasize the transformation of inputs into outputs, as illustrated in Figure 8.

The author's experience suggests that the majority of clients that outsource localization tend to conceive of translation and localization projects in much the same way, viewing the source and target materials in terms of a linear input-output relationship. In projects that require the translation of an ostensibly pragmatic, informative text such as a software GUI or Help file, it is presumed that there exists a

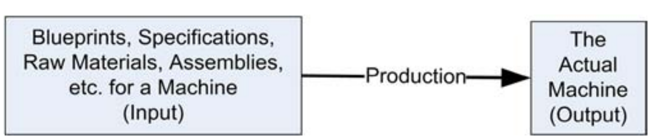

Figure 8. Linear perspective of manufacturing process.

Figure 9. Translation as transcoding.

direct correlation between what is written and what was meant. In semiotic terms, we would say that the *signifier*, or written word, corresponds to a given *signified*, or concept. Thus, clients often presume — erroneously — that translation is a matter of simply transcoding a source text into the equivalent target text (see Figure 9).

Transcoding presupposes that there is a one-to-one correspondence between the form, code, and message of the source and target texts. In this model, translation is a matter of boilerplate substitution of target equivalents for given source elements, and the translator is analogous to a machine or assembly line that stamps out target texts. Furthermore, it is presumed that translators are more or less interchangeable, along the lines of skilled machine operators, and that the translation process is a repeatable and reproducible one.[5]

Nevertheless, there are substantial risks involved in mapping a manufacturing-based paradigm of repeatability and reproducibility onto the translation process, as a simple translation and reverse translation will suffice to show. For instance, let us submit three sentences from an uninstallation wizard dialog box to machine translation from English to German using the AltaVista® Babel Fish Translation (http://babelfish.altavista.com/tr), and then back-translate the result.

English source	Press the Finish button to perform the uninstall. Press the back button to change any of the uninstall options. Press the Cancel button to exit the uninstall.
English to German translation	Betätigen Sie die Endetaste, um das uninstall durchzuführen. Betätigen Sie die rückseitige Taste, um irgendwelche der uninstall Wahlen zu ändern. Betätigen Sie die Löschentaste, um das uninstall herauszunehmen.
German to English back-translation	Press the end key, in order to accomplish university stable. Press the back key, in order to change any the university University of elections. Press the deletion key, in order to take university stable out.

If translation were merely a matter of transcoding, the repeatability and reproducibility paradigm holds that we should be able to translate a sample from source language A to target language B and back again with few (if any) discrepancies between the initial sample and the final product.

However, as the above example suggests, translation from a source language to a target language has less to do with form and code (i.e., words and structures)

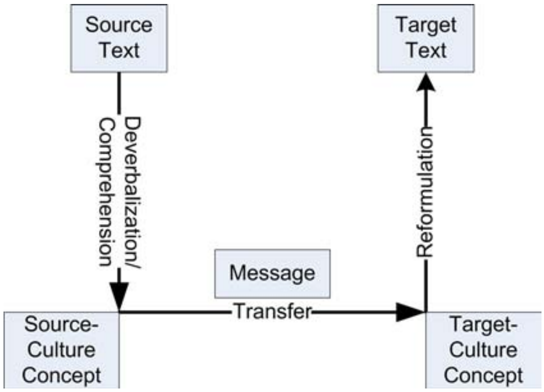

Figure 10. Meaning-based translation process.[6]

than with meanings and messages. Indeed, theorists agree that translation involves three sets of processes: comprehension, code-switching and target-text production. In the common box-diagram depiction of translation as formulated by Nida and Seleskovitch, the source text is analyzed and stripped of its specific linguistic attributes to render it into its simplest and structurally cleanest forms (deverbalization) to facilitate understanding of the presumably translinguistic message (comprehension), followed by transfer into the target language and re-encoding of this message according to the constraints imposed by target audience expectations and by the target-language grammar (reformulation) (see Figure 10).

Seleskovitch's confusion of meaning and sense notwithstanding, this simplified model is a useful window through which to explore the translation process in localization. The problem with translation in localization is that when we lack context, transcoding cannot possibly suffice to render an accurate translation. Nor for that matter can the meaning-based translation process, either. In software localization,[7] the production of a translation that fulfils quality requirements in terms of both conformance and fitness for intended use arguably requires three degrees of comprehension: (a) comprehension of the apparent literal meaning of the source text, (b) comprehension that this literal meaning may not be correct in context, and (c) comprehension of the intended meaning (see Figure 11).[8]

In Figure 11, 1–2–3–4 represents the typical translation process, 1–2a–3a–4a represents the mistranslation process, and 1–2a–1b–2–3–4 represents risk management strategy in the localization translation process. Whereas a translator would normally proceed from 1 to 2 in a classic document translation, localization often requires reformulation of the source text for a variety of reasons (such as those in the bulleted list on pages 105–106). First, the source materials may have been authored by non-native speakers, due to offshoring and outsourcing. (At the moment, this is generally only an issue for software authored in English.)

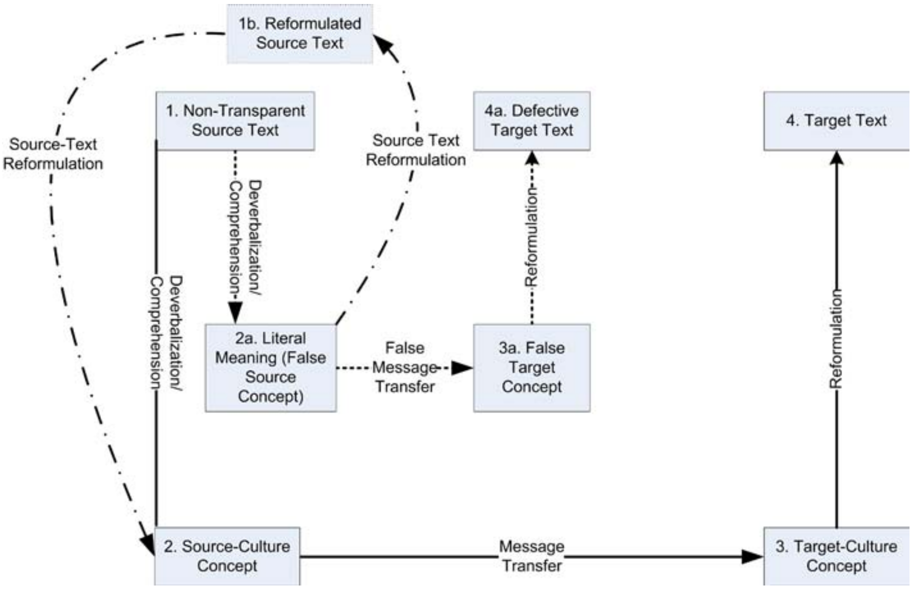

Figure 11. Meaning-based translation process applied to non-transparent source texts, or "Garbage In, Excellence Out."

Second, no attempts may have been made to develop or implement linguistic quality control processes during authoring, such as systematic terminology development and management, or style guides for the source materials. In addition, the source materials may suffer from an abundance of jargon and/or metaphorical turns of phrase that should be avoided in highly pragmatic texts, whose purpose is to inform target users rather than impress them with creativity and/or the richness of the author's language. Last but certainly not least, reformulation is inevitable during translation of the GUI when concatenation is involved.

Thus, if working in Catalyst's WYSIWYG mode (Figure 5, right-hand image), the localizer is more likely to take the 1–2–3–4 route as shown Figure 11, while if working on a *.resx file in SDLX (Figure 6) or TRADOS TagEditor, the translator is condemned to follow the longer route, i.e., 1–2a–1b–2–3–4 as illustrated in Figure 11. In the latter workflow, the risks of mistranslation are significantly higher, the time necessary to perform the tasks is increased, therefore the cost of the process is increased as well, and the project timeline may be jeopardized if proper provisions have not been made.

The aforementioned issues are due in no small measure to lack of context. However, other quality issues are raised by the lexical features of the text itself. Let us consider for example the problem of "developerspeak," namely the use of jargon and figurative or metaphorical language that has no place in denotative, pragmatic texts such as software. For instance, the verb "populate" means to supply with

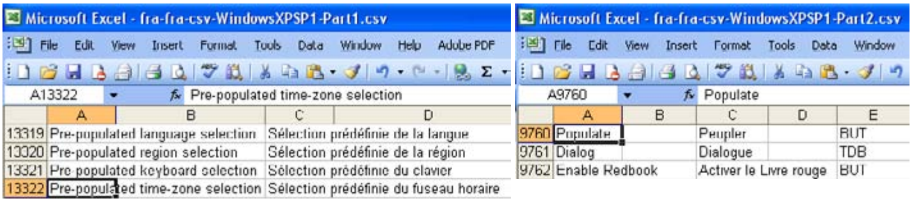

Figure 12. Translation problems caused by figurative usage: "populate" in the en-US/fr-FR bilingual Microsoft Windows XP Glossary of Translated User Interface Terms.

inhabitants, as by colonization. However, this verb is frequently used figuratively in software to refer to the process by which databases, forms, Web pages and so forth are automatically filled with data. Although the term is so widely used today that a reasonably experienced localizer or localization translator should understand the intended meaning, literal translations of the primary meaning of the term are nonetheless easily found in French localized software, to cite just one language. For example, the left-hand screenshot in Figure 12 shows a correct French translation of the term, whereas the right-hand image shows an incorrect literal translation (or transcoding, moving directly from 1 to 4a in Figure 11).

Since such metaphorical usage may impede understanding of the message by non-native speakers (i.e., translators) and non-specialists alike, it might be preferable to use a more transparent term such as "to fill" in place of "populate." This example underscores the risks of product failure associated with uncontrolled terminology (Wright 2001b: 491–492).

Computer jargon poses similar challenges. Let us consider the case of the noun "breadcrumbing." In common usage, "breadcrumbing" refers to "a navigation technique which displays a list of places a person has visited" (Maxwell 2003) (see Figure 13).

This term raises a number of issues for translators and localizers. First, "breadcrumbing" is a structural calque. The word "breadcrumb" is not a verb, but rather a noun. However it is employed as a verb in this metaphor and transformed into a present participle via the addition of the suffix "-ing." A translator who searches in a standard English-language dictionary will find no verb under the entry "breadcrumb."

Second, its meaning is culturally bound: those who are not familiar with Western European fairy tales may have no idea that the noun "breadcrumbing" plays on the story of Hansel and Gretel, who leave a trail of breadcrumbs as they are led

Figure 13. "Breadcrumbing" in The Elsmar Cove Web site forum.

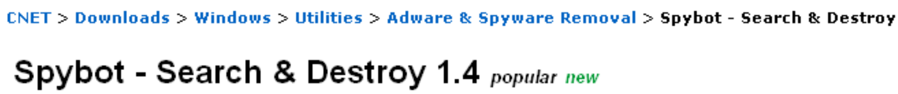

Figure 14. The "Breadcrumbing" trail in a page visited via a search engine result link.

deep into a forest in order to find their way back out. As DiFranco has observed, "this term is totally unacceptable in terms of an East Asian localizer. The time and effort it would take to translate this term into the language of the target locale would be costly" (2002).

Third, on a semantic level, there is a divergence between the connotations of the term and what it actually represents in the context of many Web site architectures. In fact, "breadcrumbing" does not generally represent the user's *path* so much as it represents the *position* of a given page within the hierarchy of a site: "Like 'You are here' indicators, Breadcrumbs show where you are" (Krug 2000: 75). Thus if we execute a search engine request for the terms "Spybot Search & Destroy" and "Download," a link that displays near the top of the results points to a CNET Download.com page from which we can download the software. The page in question displays a "breadcrumbing" trail as shown in Figure 14.

However, the breadcrumbing trail in Figure 14 is not the path by which the page was reached. Clearly in this case — and in many Web sites today — breadcrumbing does not reflect user navigation *per se* but rather user location.[9] This example underscores the fact that in practice today, "breadcrumbing" corresponds to multiple concepts. The lack of a one-to-one correspondence between term and concept undermines the ability of translators to accurately ascertain the intended meaning of the source text, and by extension, to provide a quality target-language equivalent.

Last but not least, as Hudson observes, "strictly speaking 'breadcrumb navigation' is a misnomer since Hansel's breadcrumbs were eaten by birds with the result that he and Gretel remained lost, started snacking on the witch's gingerbread house, and so on. His earlier attempt with white pebbles was more successful, but 'white pebble navigation' doesn't have quite the same ring!" (2004). Interestingly, just as Hansel's navigational strategy ultimately proved ineffective, Hull (2004) notes that "recent studies have shown that while the use of breadcrumb trails to navigate a Web site can be helpful, few users choose to utilize this method of navigation."

In sum, the breadcrumbing functionality and term reflect the perspective of developers, not that of users. Assuming that the terminologist and/or translator understand the breadcrumb metaphor in the first place, how then should they address this term? Will those who have designed and implemented the feature in the product and those who authored the GUI and documentation be aware of these potential problems of meaning? Will they be able to clearly articulate to the localization team the precise connotation of the term that they wish to convey? Will the true extent of this term's slipperiness dawn on the localization team, and if so, will they request clarification to avoid the risk of mistranslation? If so, will such clarification be forthcoming in timely fashion? Will introducing an element from a children's story into the application be seen as an unwelcome intrusion of cultural bias by users in the target locale(s)? How much time — and thus money — will be wasted by vendor and client project teams in submitting, tracking and replying to queries about this term and concept? Is the actual value provided by the functionality and terminology worth the cost and non-value-added activity that they generate?

On a larger scale, will sloppy materials and a lack of formal source-text quality assurance require the submission of dozens or even hundreds of queries to ensure that the meaning rendered in translation accurately reflects the intended meaning of the source text, which cannot be ascertained as written? How many such problem strings will be encountered during a given project? What will be the cost — in terms of both time and money — of ascertaining their precise meanings, communicating them to the team(s) and tracking the information? Who will bear the financial cost of this work in the event that it was not addressed in the project contract? What if localization into multiple languages is not simultaneous, but rather sequential? How then to manage this knowledge and deal with these issues across all languages?

ISO 9001 in localization: Neither a panacea nor a lost cause

We have seen how challenging the translation-related aspects of localization can be due to lack of context, ambiguity, as well as errors, inconsistency and problems of usage in the source-language materials. What strategies and processes can we develop to address these issues and to manage quality in a more proactive way? The answer lies in rethinking the current quality management and localization project paradigm to effectively integrate ISO processes.

Localization quality management does not begin with the handoff of source materials from the client to the localization vendor. On the contrary, quality management begins — or rather, should begin — *before the product to be localized has even been authored*. In current localization workflows, the vendor/localization teams first become involved in the project when the development of the source-language product is completed (see Figure 15).

Some would object that this sequential model is not an option when companies choose to pursue a policy of "simship," or simultaneous shipment of source-language and localized versions of products. On the surface, simship does indeed

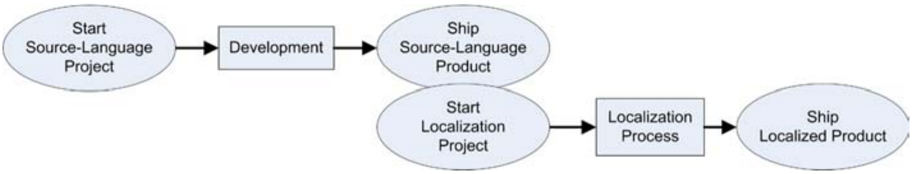

Figure 15. Typical localization project viewed in terms of the process-based quality management model (Figure 2).

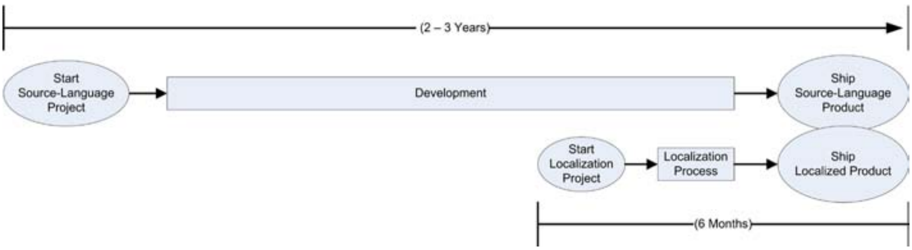

Figure 16. Typical simship project viewed in terms of the process-based quality management model (Figure 2). (See also DiFranco 53–54 and 63–64 in this volume.)

appear to make the sequential model impractical, if not impossible. Generally speaking, however, simship as practiced by many large companies today does not fundamentally alter the sequential nature of the relationship between development and localization. While the simship model does shift the localization project timeframe somewhat to the left, thus causing some degree of overlap between localization and development, localization is still by and large a tail-end process (see Figure 16).

Proportionally speaking, simship localization cycles are exponentially shorter than development cycles. Telescoping localization efforts into a disproportionately small timeframe may actually exacerbate any localization problems that arise since there is little margin for error, nor is there time to remediate internationalization flaws or other issues that may have a critical impact on the localizability of a given product.

The larger problem with the sequential model and workflow is that many clients conceive of "QA" as something for which vendors can and will assume responsibility during translation and localization. Consequently, the vendor and the localization team are made responsible and held accountable for problems that are often beyond their control. In this model, the use of the term "localization QA" is a logical fallacy when referring to the resolution of source file-related problems. Most of what passes for QA in localization can in fact never be anything other than QC since it is dependent upon and determined by characteristics of source-text receivables. Likewise, the project paradigm shown in Figure 15 effectively undermines the very possibility of repeatable quality since no standardization of

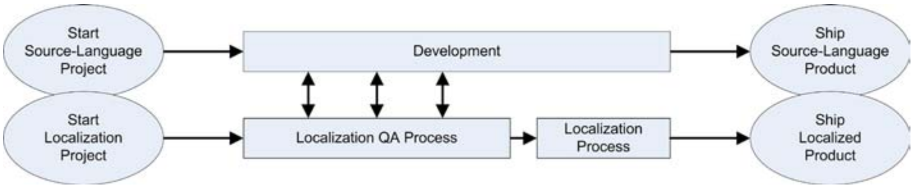

Figure 17. Quality management plan in the localization project.

localization processes can occur without standardization of client-side processes. Today, translation and localization services tend to be marketed, sold and billed as mass production whereas they are usually in fact forms of custom work.

True integration of ISO 9001 standards and quality models would require the application of comprehensive QA strategies during the *authoring* of the source materials and integration of localization therewith (see Figure 17).

In the absence of comprehensive client specifications, and in the context of *ex post facto* localization, the achievement of true quality is more of an ideal to strive toward than a goal that can be effectively achieved in practice.

Ideally, the QA strategies that should be developed and implemented for software localization to enable effective ISO 9001 quality management are the following:

- Systemic development and implementation of internationalization strategies
- Expansion of the scope of internationalization and localization QA to encompass *all* facets of authoring, including GUI content and documentation, instead of focusing exclusively on source code and functional QA
- Source-text style guide development
- Controlled source terminology development, including the involvement of target-language terminologists at the stage of the source term creation
- Systematic source- and target-language terminology management

The above measures require that the localization vendor be involved at the stage of the source product development, which almost never happens in the current language industry outsourcing model. Nevertheless, it is not realistic to argue for 100% vendor/client integration in the current market due to cost pressures, price shopping and ignorance of the issues on the part of many clients.

Consequently, localization professionals must focus their efforts on developing quality management processes and effective risk management strategies to assess, and ideally, address problems in source-text materials at the stage of inspection. This implies the implementation of a number of proactive processes as part of the project setup and pre-translation phase to mitigate downstream risk during translation and subsequent integration:

- Project data mapping (software architecture)
- Concept mapping
- Style guide creation and implementation
- Terminology management (Wright 2001a)
- Code commenting (to provide information about the function and display location(s) of strings, concatenation, length limits, etc.)

The current localization outsourcing model, in which much effort is devoted to functional QA but little or none to localization QA, is akin to putting the cart before the horse. When all is said and done, the current model implicitly assumes that functionality drives the user experience. However, the vast majority of Human–Computer Interaction, at its most basic level, is governed, and indeed made possible, by language (the use of icons and pointers in user interfaces notwithstanding). The quality of the user experience can only be as good as the quality of the language that mediates that experience. The production of quality localized products requires quality source materials, i.e., a high degree of consistency, correctness, readability, conformance and *translatability* (Bredenkamp 2004: 3; emphasis added). It is time for localization clients to put the cart back behind the horse, where it belongs, and effectively integrate localization QA into the software internationalization process. Localization quality assurance may seem expensive when viewed strictly in terms of upfront project costs, but this initial outlay is ultimately less expensive than the increased downstream localization costs including the cost of putting out fires, support costs and liability costs that ultimately ensue in the absence of effective localization QA, and which risk rising in proportion to the number of supported target languages. Quality cannot be *inspected* into a product; it must be proactively *managed* — from conception to design, production and shipment.

Notes

1. In common usage, the quotation is "I can't define pornography. But I know it when I see it." However this is actually a paraphrase; the full quotation is as follows: "I have reached the conclusion, which I think is confirmed at least by negative implication in the Court's decisions since *Roth* and *Alberts*, that under the First and Fourteenth Amendments criminal laws in this area are constitutionally limited to hard-core pornography. I shall not today attempt further to define the kinds of material I understand to be embraced within that shorthand description; and perhaps I could never succeed in intelligibly doing so. But I know it when I see it, and the motion picture involved in this case is not that" (Supreme Court 1964).

2. Localization is defined here as the process by which digital content and products developed in one locale (defined in terms of geographical area, language and culture) are adapted for sale and use in another locale. Localization involves: (a) translation of textual material into the language and textual conventions of the target locale; and (b) adaptation of non-textual materials (colors, icons, bitmaps, packaging, form factors, etc.) as well as input, output and delivery mechanisms to take into account the cultural, technical and regulatory requirements of that locale.

3. The stated goal of the proposed OSCAR GMX-Q — GILT Metrics eXchange (Quality) standard is to "specify the quality requirements for translation tasks" (LISA 2005: GMX). However, the format of this standard had not yet been defined at the time of writing. See Appendix.

4. "String" is used here to refer to text that is stored and manipulated as a group. Strings can be menu items, command button captions, dialog box titles or captions, error messages, and so forth.

5. This view of translation as a formulaic, mechanical activity is extremely widespread and explains the common misconception that translation is a commoditized service.

6. This model presupposes that the text is a self-contained entity and that a translator with sufficient source-language and target-language fluency, translation skill and experience, resources and research skills will be able to produce a quality translation. However, in many localization projects and file types, accurate translation is *impossible* without outside feedback/intervention due to the lack of linguistic and grammatical context.

7. Software localization is used here for purposes of illustration, but this point also applies to the translation of any type of decontextualized content (databases, XML, CMS/GMS, etc.).

8. Technical writers encounter similar problems when authoring product documentation or user assistance materials. See for instance Kat Nagel's reply to a TCHWR-L listserv query about developers that are non-native speakers of English, "Re: What to do?" "As a practical way for the original poster to deal with the immediate situation, I suggest that you find an empty conference room, order a couple buckets of chicken wings and a six-pack of Mountain Dew, and ask a few of the ESL developers to help you figure out how to translate what they *wrote* into what they *meant*" (http://www.techwr-l.com/techwhirl/archives/0310/techwhirl-0310-01264.html).

9. The slipperiness of this term is reinforced by the fact that it has yet another meaning in gaming, where it refers to "the process of writing game dialogue with the express purpose of keeping the player inside the 'tram lines' of the gameplay. In effect, breadcrumbing dialogue is that speech whose purpose is guiding the player towards completing goals inside the game world, such that they need never get stuck in the game" (IGDA Game Writers' SIG).

References

Bredenkamp, A. 2004. "Quality: Where does it come from and how can I get some?" LISA Global Strategies Summit, Foster City, CA, June 22. *The LISA Forums Archive*. http://www.lisa.org/archive_domain/forums/2004usa/presentations/bredenkamp.html

DiFranco, C. 2002. "Localization skills and localizability requirements." *The Gotham Translator* (Dec.): 4–5, 9.

Esselink, B. 2000. *A Practical Guide to Localization.* Amsterdam/Philadelphia: John Benjamins.

Hudson, W. 2004. "Breadcrumb navigation: There's more to Hansel and Gretel than meets the eye." *interactions*, Sep./Oct. http://www.syntagm.co.uk/design/articles/breadcrumbs.htm
Hull, S.S. 2004. "Influence of training and exposure on the usage of breadcrumb navigation." *Usability News* 6 (1). http://psychology.wichita.edu/surl/usabilitynews/61/breadcrumb.htm
IGDA Game Writers' SIG (International Game Developers Association Game Writers' Special Interest Group). N.d. "Glossary of Game Writing Terms." International Game Developers Association. http://www.igda.org/writing/WritersGlossary.htm
ISO (International Organization for Standardization). 1994. ISO 8402. *Quality Management and Quality Assurance*. Geneva: ISO. Quoted in *PMBOK*.
———. 2000. ISO 9001:2000(E). *Quality management systems — Requirements*. Geneva: ISO.
Krug, S. 2000. *Don't Make Me Think: A Common Sense Approach to Web Usability*. Indianapolis, IN: New Riders.
LISA (Localization Industry Standards Association). 2005. "GILT Metrics eXchange (GMX)." Localization Industry Standards Association. http://www.lisa.org/oscar/gmx/
———. 2005. "LISA — Frequently Asked Questions." Localization Industry Standards Association. hhtp://www.lisa.orq/faqs.html
Lommel, A. 2004. "Of windshield wipers and standards. Or, how to lose $5,000 on account of $12." *Globalization Insider* 14 (1.2). http://www.lisa.org/archive_domain/newsletters/2004/1.2/index.html
Maxwell, K. 2003. "New word of the month." *MED Magazine. The Monthly Webzine of the Macmillian English Dictionary*. http://www.macmillandictionary.com/med-magazine/May2003/07-new-word-food.htm
PMI (Project Management Institute). 2000. *A Guide to the Project Management Body of Knowledge (PBMOK® Guide)*. Project Management Institute: Newton Square, PA.
Scribe Consulting. 2003 [2003–2005]. "Scribe Consulting Quality Assurance." http://www.scribeconsulting.com/pages/qualityassurance.html
Shreve, G.M. 2000. "Translation at the millennium: Prospects for the evolution of a profession." In *Paradigmenwechsel in der Translation. Festschrift für Albrecht Neubert zum 70. Geburtstag*, P.A. Shmitt (ed), 217–234. Tübingen: Stauffenburg Verlag.
Smith, M.T. 2004. "Process mapping." *The Elsmar Cove*. http://elsmar.com/pdf_files/ProcessMapping.pdf
Supreme Court of the United States. 1964. *Jacobellis v. Ohio*, 378 U.S. 184. *LexisNexis*.
TRADOS. 2004. *TRADOS TeamWorks. Product Brochure*. N.p.: TRADOS Incorporated. http://www.trados.com/library/documents/TeamWorks/eng/Teamworks_Broch_0402_e.pdf
Wright, S.E. 2001. "Infobox 26." In *Handbook of Terminology Management*, G. Budin and S.E. Wright (eds), 873–876. Philadelphia/Amsterdam: John Benjamins.
———. 2001. "Terminology and total quality management." In *Handbook of Terminology Management*, G. Budin and S.E. Wright (eds), 488–502. Philadelphia/Amsterdam: John Benjamins.

PART 3

Game localization

Issues in localizing computer games

Frank Dietz

The localization of computer games (and video games, i.e., games played on a console attached to a TV, such as the Xbox or Playstation 2) shares many issues and methodologies with software localization in general. However, game localization presents unique challenges arising from the world-making power of games, the non-linearity of games, the often chaotic game development process, and established genre conventions.

First, a common misconception must be addressed, namely that all computer and video games are simplistic "shoot-'em-up" titles (yes, those do exist) and that translating these games therefore must be easy.[1] A related misconception is that there cannot be much material to translate, as games are all about visual effects. This is an enormous oversimplification. While an entirely action-oriented game might only contain a few conversations between characters, a simple interface and a short documentation booklet, a full-fledged role-playing game or a flight simulator may include hundreds of thousands of words to be translated. There is indeed a lot to translate in games, and the work of the translator may deal with any of the following elements: the game interface, error messages, dubbed video or audio, subtitled video or audio, mission briefings and debriefings, information files on objects in the game (such as weapons), maps, bit-mapped signs, cut scenes (i.e., the pre-made scenes that are shown after the player has succeeded or failed in achieving a certain goal), words spoken by NPCs (non-player characters), questions the player character can ask the NPCs, help files, tutorials, gameplay hints, mission editors, map editors, keyboard mapping utilities, multiplayer messages, install guides, playguides, joystick setup files, keyboard layout guides, the game credits, the text on the retail box, the readme file, instructions for patch files, and more. To complicate matters, the texts to be translated might be included in plain text files, Microsoft Word documents, Excel spreadsheets, Access databases, HTML code, source code, or be part of bit-mapped graphics. Even within the framework of the same game localization project, a translator might have to switch between various file formats and programs, including proprietary utilities developed by the production team itself. By now, most programmers know that it is better to separate the translatable text from the source code (in which the accidental deletion of even a few letters by the translator could cause the game to crash); yet games are still vast, sprawling projects that require precise attention on the part of translators and translation coordinators.

The game as a world

Games create alternate worlds whose relationship to the real world can range from extreme realism to utter fantasy. As shown in Figure 1, the highest degree of verisimilitude appears in simulations, commonly known as "sims." By their very nature, these games attempt a close approximation of the outside world, while still maintaining various degrees of ludic elements. Sports simulations, for instance, might model the rules and tactics of a sport perfectly, while allowing the player to have teams from different decades play against each other. Military strategy games might almost obsessively strive for realism in terms of equipment and unit behavior, yet open up the possibility of counterfactual outcomes, such as the Confederacy winning the battle of Gettysburg. The highest degree of realism can be found in so-called "hardcore" military simulations, e.g., *Jane's F/A-18* or *Sub Command*, which model military equipment in amazing detail (see Figure 2).

In fact, during the making of the submarine simulation game *688(I) Hunter/Killer* the development team was asked by the US Navy to leave out certain elements that came too close to classified realities.

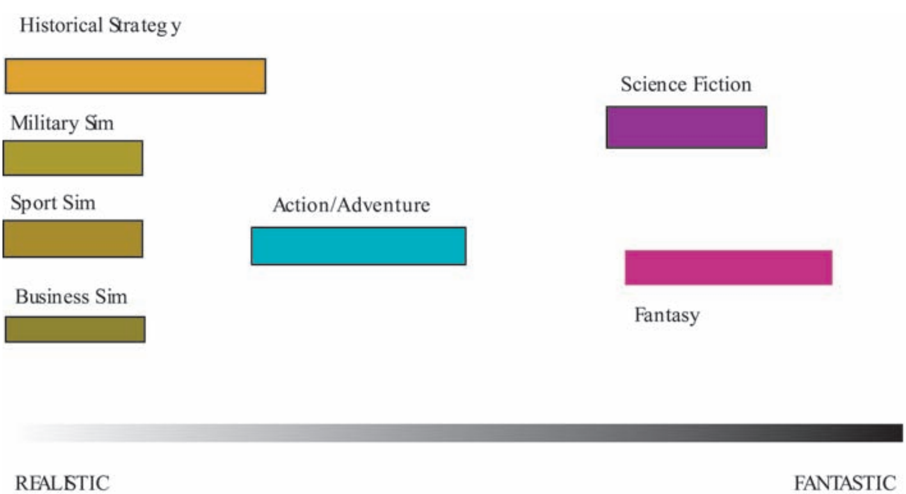

Figure 1. Realism vs. fantasy in certain game genres.

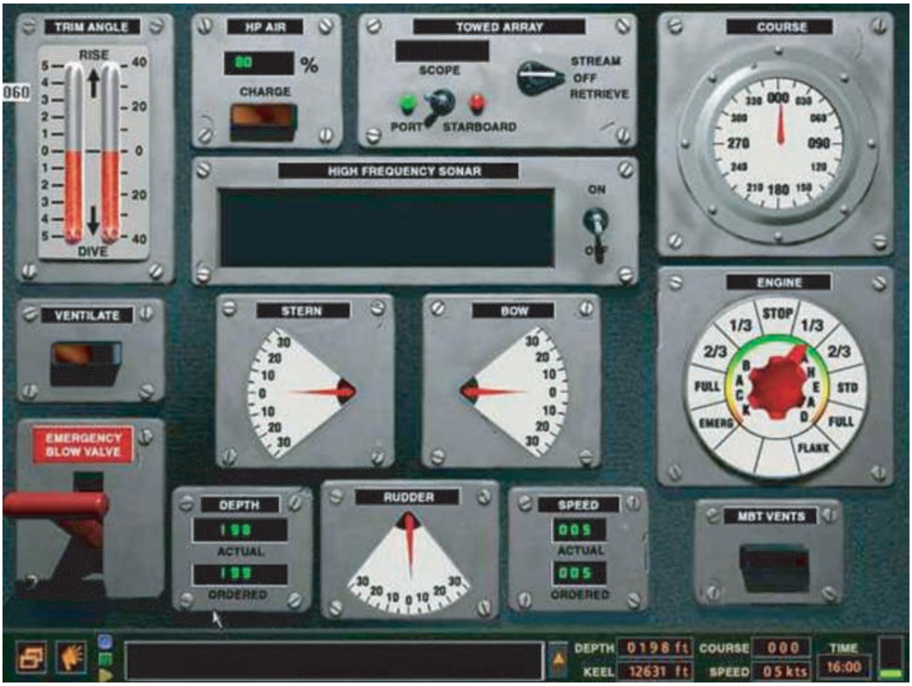

Figure 2. A screenshot from *Sub Command*.

On the other edge of the spectrum, we find fantasy and science fiction titles that create (rather than recreate) worlds of magic — in the sense of Arthur C. Clarke's well-known quote "Any sufficiently advanced technology is indistinguishable from magic."

How does the realism-magic spectrum affect localization? Essentially, it demands very different skills from the translator. Translating a highly specialized simulation, such as a flight sim, will require a high level of subject knowledge in order to handle terminology such as "active phased array radar," "imaging infrared seeker" or "thrust vectoring." In preparing for such a localization project, a translator may have to study voluminous handbooks and build up a specialized reference library. A lack of such knowledge can have embarrassing results, as in the translation of the manual for the helicopter sim *AH-64 Longbow* into Swedish — much castigated at the time on flight-sim newsgroups — that mistranslated the pilot slang expression "Winchester ammo" ("we are low on ammunition") into the statement that the attack helicopter, which is equipped with a 30-mm-cannon, Hellfire missiles and 5-inch-rockets, had only "shotgun shells" left. As one fan remarked:

> In the manual it stated that when youre [sic] wingman cannot open fire he's replying he's winchester ammo. When I now translate the swedish [sic] version it vill

> [sic] be "Sorry sir I can't fire, I got only shotgun ammo left". (Hey, wanna se [sic] the attack chopper with a twin barreld [sic] shotgun up front!) (Google groups archive for comp.sys.ibm.pc.games.flight-sim, June 15, 1996)

Sports simulations, too, require a great deal of specialized vocabulary concerning tactics and rules, as well as finding the tone of voice in which a sports commentator would describe the action.

If translating a simulation can be compared to technical translation due to its emphasis on terminological precision, the localization of science fiction or fantasy titles is more akin to literary translation. These games envision creatures, weapons, machines, entire planets completely different from the real world. The only outside reference would be to specific fictional predecessors, such as sword & sorcery novels or cyberpunk stories, for example. How does one translate "orc" or "ent" or "nano-modulator?" In the case of fantasy terms, the translator may have to dig deep in the folklore and mythology of the target culture to find an emotional equivalent to a particular mythical creature. For SF games, the translation has to maintain the appropriate level of pseudo-technological jargon and fit within the tone of the game. Therefore, on the fantasy/SF end of the game spectrum, the task of the translator involves creating a consistent tone, be it of pseudo-medieval myth or gritty cyberpunk in order to maintain the illusion created by the game.

The game as action

Games, of course, are not just worlds, but backdrops for actions taken by the player. While the so-called "back story" can lend a rich texture to the world of the game, the main functions of language in the game are to lead the player onward, to give instruction, to provide clues and to motivate. A translation will only succeed if it can replicate all of these functions.

One important factor to consider is that games can be non-linear, giving the player the opportunity to find his or her own way through the world of the game. The degree of non-linearity differs of course, and some games (such as Microsoft's *Dungeon Siege*) allow little deviation from the main course of action, while others (such as *Deus Ex* or the later entries in the *Wing Commander* series) can lead to very different outcomes, depending on the player's choices. This has consequences for translation, as a piece of information provided early on in the game might have to make sense in several different contexts.

The only way of verifying that every translation works in its context is to play the game in the localized version. This is not always easy, particularly when one considers the tight deadlines often imposed during the localization phase, but not doing so could have serious consequences. The worst that could happen (and actually has happened) is a "linguistic plot-stopper," i.e., a translation error that prevents the player of the localized version from finishing a mission or even the entire game. In the German version of the adventure game *Torin's Passage* for instance, players had to assemble "audio crystals" to pronounce a magic sentence correctly, but failed to do so, as the German translation used a different word order. The producer had to include a printed note in the game box explaining to players how to use the crystals correctly in order to continue the game (quoted in Google groups archive for de.rec.games.computer, February 11, 1997).

The development process

International markets play a crucial role for US-based game companies, and the scope of localization and the recognition of the importance of international markets have increased alongside with the growth of the computer game industry. While in the 1980s, games might not have been localized at all (or only partially, e.g., by translating only the manual and installation guide), complete localization and simultaneous or near-simultaneous launch of several language versions have now become much more prevalent.

The reason lies in the high costs of game development. Major titles can no longer be cobbled together by a few people working out of their homes, but are produced by large teams of programmers, designers, graphic artists and others who work often for years and require multi-million dollar budgets. At the same time, market competition is fierce, and the average "shelf life" of a game is extremely short — after a few months, it will be sold at a reduced price and in a year or two, you may find it in the bargain bin. All of these factors exert enormous pressure on game developers to serve multiple markets simultaneously, in order to recoup the development costs as quickly as possible.

Game localization is performed in various ways. Some companies use in-house staff, others hire a freelance translator or virtual teams of translators, or use translation agencies. In many cases, it is actually not the game developers, but the distributors or their foreign subsidiaries that take care of localization. Unfortunately, this often means that the translators will receive the material to be translated, but not the game itself.

What does this mean for the translator working on a game localization project? Unless an entire development team has had previous experience with localizing a game, the process is likely to be error-prone and difficult.

At the root of many of the problems connected with game localization lies the fact that the simultaneous or near-simultaneous release of several language versions requires parallel development. Programmers, designers, audio technicians and graphic artists who are usually under great stress during the beta and final stages of game development will also have to devote some of their time to creating (and fixing) foreign-language versions. On the translator's side this parallel development means working with a text that is, despite all assurances to the contrary, still fluid, and sometimes requires frantic re-writing and re-translating during the last few days before the game's ship date.

Genre conventions and localization

While localizing a particular game, localizers should pay attention to certain genre conventions for the target market. This could include certain terms that might not have to be translated. In German versions of role-playing games, for instance, the term "NPC" ("non-player character") is usually left untranslated. Translators also have to be aware of genre-specific terminology, such as "real-time strategy," "turn-based strategy," "cooperative multiplayer" or "deathmatch." Translators who localize computer and video games should also read computer game magazines such as *PC Gamer* in their target language or access relevant Usenet newsgroups in order to stay current with genre-related terminology.

The importance of the interface

Interface design is a difficult issue in any form of software localization, as word lengths can differ greatly between languages: a classic example is the English "Quit" vs. the German "Abbrechen." In this case, translation causes the text to expand from 4 to 9 characters. Yet game localization adds further complications to the issue of interface design and localization, which are related to the dual nature of a game as world and action.

First, a game interface should not destroy the player's willing suspension of disbelief concerning the "reality" of the game. Unlike the interface of a normal application, which is integrated into the program, it exists as a quasi-transparent layer between the world of the game and the world of the player. Therefore, it must be both unobtrusive and fully functional. As shown in Figures 3 and 4, the game interface usually blends stylistically into the look and feel of the game, with a fantasy role-playing game like *Ultima Ascension* sporting earth tones, while a futuristic game has a more hard-edged, high-tech appearance. In both cases, though, the interface is pushed to the margin of the screen, which limits the amount of text that can be displayed. In the case of *System Shock 2*, even the English version

Figure 3. Icon-based interface in *Ultima Ascension*.

Figure 4. The crowded inventory of *System Shock 2*.

Figure 5. Use of localized tooltips in *1503 A.D.*
© Sunflowers.

already uses abbreviations for labeling interface elements, which can (and did) create considerable problems in localization.

The seemingly obvious solution, the use of symbols, is not without problems, as they are not always culture-neutral. I remember translating a flight simulation game in which the function "Go the first page of the interface" was visualized through the icon of a baseball home plate — an image that would convey nothing to the vast majority of German players, and unfortunately could not be changed due to tight deadlines. Furthermore, symbols convey limited messages. While objects (gold, stone, wood), simple actions (fight) or basic concepts (health) can be depicted relatively easily, more complex actions (save game and quit to the desktop) might not be able to be conveyed via a pictogram. A common solution is the use of mouse-over labels (called "tooltips" in Windows parlance) that appear when the player moves the mouse cursor over an icon (see Figure 5).

A special case of interface applies to modern flight simulation games, i.e., military jet and helicopter sims. In flight simulation games, a so-called HUD (heads-up display) is projected onto the center of the screen and displays information about altitude, angle of attack, airspeed, selected weapons and much more. This closely models the real HUD projected onto the canopy of a military aircraft.

Interestingly, most German fans of the *Jane's Combat Simulations* series preferred the HUD to remain in English (it had been translated in an early game of the series, *Advanced Tactical Fighters*), in order to preserve the sense of realism so cherished by hard-core flight sim gamers. This immersion into the virtual cockpit even goes so far that some players spend hundreds of dollars on highly realistic flightsticks, throttles and rudder pedals in order to create a true HOTAS ("hands on throttle and stick") flight experience.

Warning messages in flight simulations present similar challenges. In order to preserve the sense of realism, these audio messages are often not translated, but rather provided with subtitles. As games are always actions, game messages must be time-sensitive. Unlike what happens in more utilitarian software, which allows the user plenty of time to react to messages, warnings in games often require immediate responses. This is particularly important if the message takes the form of English audio with a target-language subtitle. The player should not have to spend too much time reading that, for instance, a surface-to-air missile is approaching his or her plane.

Cultural issues

In some cases, a game cannot be simply translated and released in the target culture, but must first be "culturalized," i.e., adapted to account for certain cultural conventions and preferences:

> Games are created for entertainment value and therefore tend to be carefully crafted to suit a particular audience … Western game characters, for example, are usually more "adult-like" (think of He-Man or Lara Croft), while Asian characters typically emphasize more child-like characteristics (think of Japanese *anime* or *manga*) and have more of a fantasy look and feel. For certain titles, localizers may find they need to recreate characters for the target market (Trainor 2003: 18).

A related issue is the depiction of violence and/or sexuality in games. While violence is of particular concern on the European market, nudity is much less so. Indeed, Pham and Sandell quote a spokeswoman for the German agency *Bundesprüfstelle für jugendgefährdende Medien* as stating that "the representation of unclothed people is not relevant in the sense of the protection of children and young people." Computer and video games have come under intense public scrutiny in several European countries. In Germany, for instance, the press linked a teenager's 2002 shooting spree at an Erfurt school (which left more than a dozen people dead) with his frequent playing of the first-person shooter game *Counter-Strike,* and several politicians called for a general ban of "killer games." Germany has tightened

Figure 6. Violence in the US version of *Crusader: No Remorse.*

existing laws concerning the depiction of violence in computer and video games, and there is a movement towards a European legislation in this area.[2]

The effect of these laws is that games developed in the US often receive higher age ratings (a game that is labeled "Teen" in the US might be rated 18+ in a European country), and that marketing activities for extremely violent games may be severely restricted or even banned outright (see Kreimeier). Such restrictions, of course, can clearly limit the sales of a game in an important market, so game companies often adapt localized versions to adhere to national rating systems. Figure 6, for instance, shows a scene from the game *Crusader: No Remorse* in which the character controlled by the player (in the center) incinerates an enemy who screams while he is dying. Throughout the entire German version, this scene (and similar ones) was replaced with the victim just falling down, without bursting into flames or screaming.

In other cases, developers or their distributors have replaced red blood with green, introduced adjustable "gore settings" (sometimes with password-controlled parental lock-out functions) or even replaced human opponents with robotic ones (with considerable repercussions for the plot) in order to avoid legal restrictions: "In the original 'Command & Conquer' game, enemy soldiers were turned into robots in the German version. Instead of blood, they spurted oil. In 'Grand Theft Auto: Vice City,' victims in Germany don't bleed and body parts can't be severed" (Pham and Sandell 2003).

This, in turn, has created a movement among fans to reverse these changes in the localized versions by means of so-called "bloodpatches," i.e., small programs that unlock the blood and violence levels present in the US version of the game.[3]

Aside from violence, national ratings boards also focus on sexually explicit content, abusive language and depictions of drug use in games. The University of Oxford's Programme in Comparative Media Law and Policy has produced a whitepaper entitled, "Electronic Game Industry Self-Regulation: Comparison of American ESRB, British VSC and Dutch NICAM Codes," which provides a good overview of various European approaches to these issues.

In other cases, cultural references may have to be changed to make the game understandable for the target market. In their article "Nudity in games OK, but blood verboten in Germany," Pham and Sandell report that Nintendo of America spent six months on

> converting a single Japanese game, "Animal Crossing," for the U.S. market. Hundreds of characters in the game had to be given new names. Holidays that were peculiarly Japanese, such as White Day on March 14 when Japanese girls give gifts to boys, were rooted out and American holidays were added, including Thanksgiving and the Fourth of July (Pham and Sandell 2003).

Figure 7. Photo of a He-111 without swastika on fin.
© Jane's Combat Simulations.

Cultural changes may also be necessary when localizing historical games, as was the case during the German localization of *Jane's WW2 Fighters*, a flight simulation set in 1944. In order to comply with German laws prohibiting the display of the Nazi swastika, the developer not only replaced the swastikas on the fins of in-game airplanes, but also electronically erased them in various examples of historical footage provided as background to the action of the game (the Battle of the Bulge), as can be seen in Figure 7.

Conclusion

Game localization can be a complicated and chaotic endeavor. The structure of the gaming industry, the often rather parochial outlook of development teams, and the enormous pressure to ship several language versions of a game at once (particularly in publicly traded game companies) all create obstacles for the localization process. There are a number of steps, though, that both translators and members of development teams can take to reduce friction and make the process more effective:

- There should be early and frequent communication between translators and developers in order to avoid interface design dilemmas, file format issues or cultural insensitivities.
- Translators should also have one contact person in the development team (sometimes jokingly referred to as the "translation czar/czarina") who can distribute queries to the appropriate team members and ensure that all relevant materials are routed to the translators.
- Source code tracking software (such as Visual SourceSafe) should be employed not only to track revisions of code, but also to ensure that changes in the English text are flagged and communicated to the translators.
- Translators should receive basic design documents early on, so that they can gather reference material suitable for the particular type of game.
- The use of translation memory tools, which is particularly important considering that many successful games have several sequels and add-ons, should be increased. This could involve an industry-standard program like TRADOS, or a proprietary product such as Ion Storm's LÖGAN.

Most importantly, translators should have a chance and be expected to play the games they are localizing. "Blind" localizations are unfortunately still all too common, partly due to developers' concerns about software piracy, though protection utilities such as SafeDisc reduce that risk, and partly because of lack of interest on the part of translators who may not be aware of the complexities of game localization. The (decidedly) second-best solution is to provide translators with a wealth of background information, such as screenshots, design documents and game walkthroughs and later have the game tested by native speakers of the respective target language.

Notes

1. While I was working as an in-house translator at Origin Systems, we actually received an application letter that stated, "I have never before translated computer games, but how hard can that be?"

2. For a discussion of the "Pan-European Game Information system or PEGI, see the Web site of the Interactive Software Federation of Europe at http://www.isfe-eu.org.

3. For a German glossary of bloodpatch-related terms, see http://blutpatches.gamerstalk.de/index.php?site=glossar.

References

Bates, B. 2001 *Game Design: The Art and Business of Creating Games*. Rocklin, CA: Premier Press.

Bemis, G. 2002. "Game localization: What does it take to bring foreign games to the US?" *TechTV,* January 18. http://www.techtv.com/extendedplay/videofeatures/story/0,24330,3368070,00.html

Dietz, F. 1999. "Beyond PacMan: Translating for the computer game industry." *ATA Chronicle* 28 (9): 57.

———. 2002. *Software Localization Glossary*. http://www.frankdietz.com/softgloss.htm/

———. 2003. "A translator's perspective on games localization." *MultiLingual Computing & Technology* 14 (5): 21–25.

Esselink, B. 2000. *A Practical Guide to Localization*. Amsterdam/Philadelphia: John Benjamins.

King, B. and Borland, J. 2003. *Dungeons and Dreamers: The Rise of Computer Game Culture from Geek to Chic*. New York: McGraw-Hill.

Kreimeier, B. 1999. "Killing games: A look at German videogame legislation." *Gamasutra — The Art and Science of Making Games*, August 27. http://www.gamasutra.com/features/19990827/killing_games_printer.htm

Larsen, S. 2002. *Playing the Game: Managing Computer Game Development*. International Edition, Version 1.1. Blackwood Interactive. http://www.blackwood.dk/postmortem/PDF/PlayingTheGame-IE.pdf

Mandel, B. 2000. "Unraveling the mysteries of game localization." *The Adrenaline Vault*, March 10. http://www.avault.com/articles/getarticle.asp?name=local

Pham, A. and Sandell, S. "Nudity in games OK, but blood verboten in Germany." *Los Angeles Times*, sec. C, June 9, 2003.

Trainor, H. 2003. "Games localization: Production and testing." *MultiLingual Computing & Technology* 14 (5): 17–20.

University of Oxford Programme in Comparative Media Law and Policy. "Electronic game industry self-regulation: Comparison of American ESRB, British VSC and Dutch NICAM codes." Selfregulation.info. http://www.selfregulation.info/iapcoda/games01-study-020716..zip

Localizing MMORPGs

Eric Heimburg

Massively-Multiplayer Online Role-Playing Games (MMORPGs) are a particular variety of video and computer game that is growing in popularity around the world. MMORPGs distinguish themselves from other types of games in several fundamental ways. In a MMORPG, hundreds or thousands of players interact with each other, and share resources, difficulties, and activities in the same virtual game world. These games are also known for not having a definite ending — there is no way to "win" a MMORPG. As a result, players can play the same game for thousands of hours. This open-endedness is a key to the success of MMORPGs, as most of them are based on a subscription model; these games typically charge \$10–\$20 US dollars per month, in addition to the purchase of the game itself. To keep the players interested (and paying the monthly fee), most creators of MMORPGs periodically add new activities and content to the game.

By their very nature, MMORPGs present a number of unique localization challenges. First, certain issues arise when people try to play the same game in several languages simultaneously. Localization of MMORPGs is also more difficult due to the sheer amount of text they contain, and their ever-expanding nature — they are constantly updated with new text throughout their multi-year lifespan. Finally, perhaps the most daunting challenge in the localization of MMORPGs is ensuring high quality grammar and accurate translations in all supported languages.

One game, many languages

One of the first decisions that must be made when localizing a MMORPG is whether or not it will be possible for speakers of different languages to play on the same world, side by side. If multiple simultaneous languages are simply disallowed, every supported language will require a separate game world, but many technical difficulties will be avoided. However, a more full-featured MMORPG would allow multiple simultaneous languages, and players are coming to expect this feature. Nevertheless, enabling multiple simultaneous languages raises a certain number of localization-related issues.

First, there is the issue of player names: which letters and symbols will be valid in these names? Most MMORPGs let you communicate and interact with other players by typing their name; the name provides a unique way to differentiate between characters. However, what happens when you cannot type the symbols that make up someone else's name? You will be unable to talk to that person, but the person will be able to talk to you. In addition to allowing dialog between players, most games also offer a variety of other services that are keyed to a player's name, such as ignore lists (so you don't have to see the text of rude players) and friends lists (so you can always tell when your friends are online). If you can't type the name, you can't use these features.

The lowest common denominator is usually English, because everybody using a computer can type names that contain the English letters and numbers. However, the French keyboard, for instance, allows one to type many accented characters, such as è or à, in addition to the standard Latin characters used in English. The German keyboard, too, has keys for Ü, Ö, and various other accented characters that people with English keyboards can't easily type, and Asian keyboards allow the user to type thousands of symbols via special software available only in Asian versions of games. So the simplest — though least flexible — solution is to allow only English-letter names: all keyboards at least allow those letters to be typed. However, this approach is not ideal, especially for Asian players.

MMORPGs struggle with this issue in various ways. Typically a combination of solutions is adopted. For instance, a game may allow all US and European players to play together on the same world, but require that they use English names, whereas Korean users play on different worlds entirely, and are allowed to use their native characters for names.

To avoid this difficulty, some games do away with the need to type player names at all. They provide other interfaces for all their game features so that it is never necessary to type a player's name. Depending on the game and the features in question, this solution sometimes works well, and sometimes is very awkward. But even when it works well, this solution doesn't address the culture gap involved. For instance, US English players cannot readily tell the difference between ì, i, and í, let alone between complex Chinese characters. This can cause confusion by allowing the creation of different names that may appear to many users to be identical — but that are not (such as "Eric" and "Eríc," for example). Mischievous players take advantage of this confusion. If "Eric" is a popular player of the game, mischief-makers can name themselves something very subtly different, such as "Eríc," and pretend to be the original player. They can also attempt to ruin the original player's good name by acting inappropriately, or try to trick the original player's friends into giving them free items.

For this reason, perhaps an ideal solution is to allow names that include variants of Latin characters (such as í), but to internally treat them as identical to the corresponding English letters (such as i), except when drawing them to the screen. Thus, "Eríc" would be treated the same as "Eric" in every way except when being displayed to the user. This would prevent the creation of a character named "Eríc" if there was already a character named "Eric" — the game would consider the new character to have a duplicate name, and disallow it. This approach would also let players use special characters in their names without preventing other players from typing the name on non-native keyboards. If someone saw a player named "Höfud," for instance, they could communicate with that player by typing the name as "Hofud" — the game would correctly identify the player, since there could be no ambiguity. This is a good approach for English and many other Western languages. However, it only addresses Latin characters and does not help European players differentiate similar-looking Korean names, for instance.

Beyond the issue of player names, MMORPGs pose a much broader question: how can people who speak different languages communicate at all within the game? Different games adopt different approaches. The most common one is to have no specific solution at all: in such cases, when a player speaks in German, everyone sees the German words, regardless of whether they speak German, or French, or some other language. This approach only permits communication between players to the extent that they share a common language. (Even this non-solution may require some extra coding. For instance, displaying Korean characters on an English computer sometimes requires special handling.)

Many MMORPGs take things a step further, and actually try to facilitate communication between players who speak different languages. Some games, such as Electronic Arts' *Ultima Online*, provide users with in-game translation software. (In the case of *Ultima Online*, Systran provided the translation systems). When used, this feature automatically translates other languages into the player's native tongue, so that when a user types something in German, for example, French users see it in automatically-translated French. Such software is usually adequate to get general concepts across, but many users feel that it is not sufficiently advanced to convey complex concepts — when elaborate sentences are typed, the resulting translations are often comically inaccurate, and can lead to misunderstandings or hurt feelings. However, as translation software advances over time, this may become an increasingly adequate solution.

Another option is to provide menu-based communication systems. Games that provide this option let players build sentences by choosing basic sentence structures and then filling in the blanks with subjects. This option is especially convenient for console games, because consoles such as Sony's PlayStation 2 typically do not have keyboards attached to them. (Keyboards are sold separately.) When a game provides menus for communicating, players can navigate these menus using their regular game controller, and don't have to rely on a keyboard. Once a user has completely built a sentence using the menus, that sentence is then sent to the recipients, each of whom sees the sentence in his or her own native language.

Menu-based talk trees provide very good translation, because only certain sentences are possible, and the designers can make certain that all possible combinations can be translated correctly for every language. However, the very thing that makes menus powerful — their limited choice of sentences — means that only simple concepts can be conveyed in this manner. Many users still feel the need to use a keyboard to type complicated sentences, which bypasses the automatic translation feature of the menus.

SEGA's popular home console video game, *Phantasy Star Online*, provides a menu-based communication system, which allows players to discuss many aspects of the game to a reasonable extent. It also goes a step further and lets players draw their own custom pictographs. The player can display these pictographs to other nearby players at any time. Some players draw iconic faces with this system, so that they can instantly convey sadness or happiness in a language-independent way. The only problem with this feature is a minor detail of the implementation: the pictographs take up a large part of the screen for everybody who sees them, so players often ask their friends not to use too many pictographs, as it can distract from the game. This problem could be corrected in a more nuanced implementation of the idea.

Localization strategies must address not only communication between players, but also communication between the game and players. Such communication usually takes the form of printed text (as opposed to, say, audio or movies) because printed text requires little bandwidth when the developer wants to send new content to players. Generally speaking, this text is localized in an MMORPG as it would be in any other video game. However, there are a few considerations specific to MMORPGs.

Managing the text

The amount of text in MMORPGs is greater than that of an average game of another genre. An MMORPG can easily contain hundreds of thousands of words of text. Turbine Entertainment's *Asheron's Call 2* totals some 350,000 words at the time of this writing, with an additional 15,000 to 20,000 words being added every month. Older games have even more text, which they have accumulated over the course of many years.

The text of a localized MMORPG is stored in the form of "strings." In programming terms, a string is text that is stored and manipulated as a group. A string might be a sentence, a paragraph, or just a single word — it can be any amount of text. The size and contents of a string depend on its context, function and meaning. For instance, the string for a button label might consist of a single word, such as "OK," whereas the string for a subtitle in a cut scene might be a long speech by a game character. Every bit of text in a localized game is stored as strings of one size or another. These strings are stored in tables for translation. A "string table" is a collection of strings in some format that both the translators and the programmers agree upon.

When designers decide how to store the string tables, they must take into account an interesting social phenomenon. Most gamers are competitive, and MMORPG gamers are no exception. In a game such as an MMORPG, one form of competition is based on seeing who can be the first to complete a new quest. This gives a player bragging rights over others who are also playing on the same world. In order to enhance their chances of being first, many players resort to cheating and use data manipulator tools to reverse-engineer the string tables on their computer in order to figure out how a quest works. Armed with this information, they can often complete a new quest in a matter of minutes or hours, whereas the designers might have expected it to take days.

The solution to such "shortcuts" is to take advantage of the client–server nature of MMORPGs, and not provide strings to the players until they are actually needed. The program on the player's computer is called the "client," and it connects to a "server" computer. (Actually, the server typically comprises many servers linked together to provide sufficient computing power, but they act together as one entity.) A player can only see and reverse-engineer things that are on his or her own computer. Thus, by storing string tables and keeping strings hidden from the client until it actively needs them, the server prevents players from taking shortcuts that would otherwise undermine the integrity of the game.

In its simplest expression, the server merely sends strings to the client whenever the client needs them. For instance, if a player completes a portion of a quest, the server will send the exact text that the client should display to the player, in the player's chosen language. The strings are still stored in tables — but these tables are located on the server, rather than on the client.

The client–server approach can be optimized in various ways. Many strings must be displayed repeatedly while the game is played, and it is inefficient to transmit these strings over the network every time they are needed. Because network bandwidth is expensive, many games adopt a hybrid system in which the most commonly-used strings are stored in a table on the client, whereas "special" strings, such as those relating to quests, are stored on the server and are not sent unless specifically requested by the client.

Network bandwidth can be further optimized by using a client-side string cache. In other words, instead of sending strings directly to the client, the server instructs the client to display a particular string ID ("Client, show string #537 to the player."). The client then determines whether it has seen that particular string before. If so, it simply displays the string to the player immediately. But if the client has not previously seen that string, it requests it from the server ("Server, please send me string #537."). The server responds by supplying the needed text ("String #537 is 'You have completed the Chaos Ascension Quest' "). The client then displays the string, and remembers it for future use. In this way, if the client encounters this string during subsequent game play, it will not need to ask the server to supply the string value again.

A client-side cache yields the best bandwidth savings, but also creates complications in the event that any text is modified. If a string changes, the client will not realize that it has changed. It is thus important that the server notify the client in the event that such-and-such string is obsolete, so that the client will know that the string needs to be re-requested and re-sent by the server when it is next needed.

Though solving technical issues such as these can be time-consuming, they are not the hardest problems faced when localizing a MMORPG. A more daunting challenge lies in finding methods to ensure high-quality grammar and accurate translations.

Improving grammar quality with a meta-language

In a text-heavy game (i.e., most MMORPGs), players are going to be reading text for many hours. In these situations, it often pays to have good grammar. Players might not even notice grammar when it is correct, but they will certainly notice when they see poor grammar over and over for hundreds of gaming hours.

The crux of the problem with grammar in MMORPG localization is that string tables generally do not provide a lot of flexibility for good grammar. They tend to be fill-in-the-blank solutions. For instance, the following string could be found in a table of nearly any MMORPG:

```
"You killed the $MONSTER$!"
```

During play, the game client would replace the variable `$MONSTER$` with another string containing the actual name of the monster killed. This works well if the name of the monster is "goblin" or "dragon," but what if the monster has a proper name? What if the player has just killed George The Bad, the lord of all demons in your game universe? When the player kills George The Bad, the game will announce:

```
"You killed the George The Bad!"
```

Obviously (to us), that extra "the" is incorrect. Let us consider another possible string:

```
"You hand the $ITEM$ to $NPC$. He shakes his head and hands it back."
```

This string might be used with a game's NPCs (non-player-characters), the computer-driven actors in the game that interact with the players to tell stories. This particular string might be presented whenever a player tries to give a random item to an NPC, but the NPC doesn't want that item. The pronoun "he" presents the first problem: what if the NPC happens to be female, or an automaton? We're also assuming that `$NPC$` is a proper noun, and that `$ITEM$` is NOT a proper noun. These assumptions could be problematic, as shown by the following contrived misreading:

```
"You hand the Excalibur to seamstress. He shakes his head and
hands it back."
```

In theory, replacing the variables in the above string should have produced the following sentence:

```
"You hand Excalibur to the seamstress. She shakes her head and
hands it back."
```

Clearly, classic string tables and fill-in-the blank variables do not facilitate accurate grammar. So how do we correct this problem? The traditional game approach involves two steps: first, reorganizing the string to minimize the number of special cases; and second, creating an extra string for each variant of the remaining special cases. Thus, it would be necessary to create one variant of a given string to be used when the NPC is a man, one to be used when the NPC is a woman, and one to be used when it is a robot. This approach does not solve the dilemma of NPCs whose names aren't proper nouns, or items that DO have proper names. If we made variants for all the possibilities here, we'd end up with twelve different variants for just one string. Worse, additional variants would probably be required for other languages. All these extra strings entail extra programming time, extra translation costs, and extra bandwidth needed. It just doesn't make sense to have twelve variants of a string. This conclusion brings us to the implicit third step in the traditional approach: ignore bad grammar whenever the solution is deemed too difficult or costly. This tends to happen most often in the special cases, rare situations that aren't the norm for a game. If 99% of the NPCs are female, we may just use the female pronoun everywhere. If all but one of our items are singular, we might not bother specially handling the one item that is inherently plural. However, ignoring these special cases makes them glaringly obvious to players.

Compounding the issue, game designers very rarely consider that their strings may or will be translated into other languages. This means that even when they take care to polish their strings in their native language, the translated versions are poor. For instance, a common problem with games that were originally coded in English is that the developers did not consider the possibility that items might have gender. So a string that says:

```
"You pick up the $ITEM$."
```

Ends up being translated into French as:

```
"Vous avez pris la/le $ITEM$."
```

Because the translators have no way of knowing whether the variable `$ITEM$` will be replaced by an item of masculine or feminine gender, they must include both forms of the definite article, or use one and run the risk that the choice will be incorrect. Needless to say, neither of these alternatives will produce quality results in the target language.

A better approach to handling issues of gender and number is to use a simple set of codes embedded in the string itself. I call this set of codes a "meta-language" because it allows the author to embed many grammatical aspects of a language into a given sentence.

To illustrate the meta-language, let us return to our earlier example and examine what it would look like after applying a meta-language to it:

```
"You hand #1:{the[!n]} #1:$ITEM$ to #2:{the[!n]} #2:$NPC$.
#2:{He[m]|She[f]|It} shakes #2:{his[m]|her[f]|its} head and hands
it back."
```

This version is considerably more intimidating! However, this string encodes all the relevant English variations. When translated into French or German, the meta-language can be used to supply all the possible permutations of the variable string elements in those languages.

Let's look at how a meta-language works. It is basically language-agnostic, i.e., aspects of language aren't hard-coded into the system. Instead, it takes advantage of the method by which string tables are commonly laid out. Many of the strings will be nouns, and the other strings will be sentences or paragraphs into which those nouns are inserted as plug-in variables. The meta-language works by embedding data into the noun strings, so that the sentence and paragraph strings can use that data.

For the meta-language to work, each noun string must be properly marked with tags indicating which grammatical rules apply to that item. This is accomplished by appending a few special letters to the end of the noun. (These special characters are removed before the string is displayed to the player).

Fortunately, tagging nouns is easy to do. Here are some sample strings for various nouns and names, with appropriate meta-tags embedded in them:

`"avocado[v]"`

The `[v]` indicates that "avocado" starts with a vowel, so it should begin with the "an" form of the indefinite article instead of "a." The following example contains multiple tags:

`"Bob the Blacksmith[nm]"`

The `[n]` here indicates that "Bob the Blacksmith" is a proper noun, so it should not ever be prefaced with the definite article "the," for instance. The `[m]` indicates that Bob the Blacksmith is male. Consider another example:

`"Asheron[vmn]"`

This example's tags indicate that Asheron is a proper noun (`[n]`), a male name (`[m]`), and begins with a vowel(`[v]`).

The meta-tags that Turbine Entertainment used for *Asheron's Call 2* are listed below:

`[m]` = male ("he" or "him")
`[f]` = female ("she" or "her")
`[i]` = inanimate or gender-neutral (an "it")
`[p]` = plural name (as in "those pants")
`[v]` = starts with a vowel (so use "an" instead of "a")
`[n]` = name (proper noun — don't use "an," "a," or "the")

Even if a name contained in a noun string doesn't require any of these tags, it is still useful to append empty brackets `[]` to the end of the item's name, so that translators can tell at a glance which strings are "noun" strings, as opposed to paragraph or sentence strings into which those nouns are inserted. In this way, the translators can make sure to examine each noun and apply appropriate tags in the target language, even if no tags were needed in the original. For instance, the English string "`champion's long sword[]`" requires no special meta-tags in the source language. But we still append the empty "`[]`" to the end of the string anyway. This enables translators to know with certainty that this particular string is a noun, and should be given meta-tags appropriate to the translated language. When this string is translated into French, it becomes "`Épée longue de champion[fv]`." Meta-tags are present in the translated version, but not in the original. So the "`[]`" in the English version is merely a cue to inform translators that they may need to add tag information for these strings when translating.

If a particular language needed more tags, it could appropriate more letters for special cases unique to that language. For instance, Serbian has two types of plural, one for numbers ending in 1–4, and a different one for numbers ending in 5 or more. A Serbian translator could use `[p]` for noun-strings representing plurals of the first type, and create a new tag, say `[5]`, for plural nouns of the second type.

Whenever possible, it is best that the string authors add the tags to their strings themselves as they write them. It is tempting to try to write software to automatically assign meta-tags, but automatic assignment is just too error-prone. How would a computer know that "herb" should be treated as if it started with a vowel in US English ("an herb")? Only a native speaker would be able to tell that at a glance.

However, there is a category of strings to which designers cannot add meta-tags themselves, namely player names. Since players create their own names, nobody can pre-assign meta-tags to them. Consequently, the game must apply meta-tags to player names programmatically. In theory, there is a chance that programmatic application of meta-tags could cause errors, but fortunately the languages into which we translated *Asheron's Call 2* (French, German, and Korean) use few applicable rules for names, thus minimizing the possibility of error. The game automatically appends `[m]` or `[f]` to the player's name based on whether they've created a male or a female avatar. It always adds `[n]`, because the player's name is assumed to be a proper noun. And finally, it looks to see if the name starts with an English vowel, and if so, it adds `[v]`. The last rule really isn't necessary in English — after all, I'm just "Eric," not "an Eric." My name is a proper noun, which trumps the vowel rule. So it doesn't really matter if the game applies `[v]` to some names that don't need it, or vice versa.

Because we allowed only simple naming conventions in *Asheron's Call 2*, the rules were rather easily automated for English, French, and German names. Other languages might have been harder to automate. A more unusual game design might have also created problems. Imagine a game that allows players to be robots of many different types. Some players might give their robots gender-specific names ("Sally the Robot"), while others might give their robots gender-neutral names ("XT8-42"). In any event, the game's designers should devise a way to request the needed tagging attributes from players in the event that the game is unable to automatically discern the information.

Once all the "noun" strings — i.e., items, creatures, and other nouns — have been tagged, they can be used as variables in paragraph and sentence strings that use the meta-language. Paragraph and sentence strings use a meta-language to enable several different alternate wordings. When the variables of the "noun" strings are inserted into the paragraph or sentence strings, the meta-language enables the generation of grammatically correct sentences thanks to the meta-tags that indicate the proper definite article, indefinite article, or possessive adjective to be used with each variable in the target language. In Turbine's meta-language, the various permutations of definite/indefinite articles, possessive adjectives, etc., are enclosed in `{}` symbols, each option being separated by a `|` symbol. Each possible option is associated with a different meta-tag, and one can specify a "default" choice to be used when none of the given options is appropriate:

```
"I want {an[v] | a} $FOOD$."
```

This sentence can become "I want a pear," or "I want an avocado," for example, depending on whether the value of `$FOOD$` has the `[v]` tag or not. Notice that the indefinite article "a" has no tags. This lack of tags indicates to the system that "a" is the "default" choice, to be used when no other option is appropriate. In this simple example, "`an[v]`" will only be used when the `$FOOD$` noun variable contains the `[v]` tag. In all other cases, the default choice "a" will be used.

One can also indicate that a certain word or phrase is to be used only when a tag is *not* present by placing an exclamation point in front of the tag letter:

```
"You kill {the[!n]} $NAME$."
```

This would only insert "the" into the sentence if the `$NAME$` variable does *not* contain the `[n]` tag. So if the `$NAME$` variable is filled in with "`Timmy[nm]`," the resulting sentence is, "You kill Timmy," while if `$NAME$` is filled in with "`Evil Wizard[]`," the sentence becomes, "You kill the Evil Wizard."

The above examples illustrate how the presence or absence of a given single tag indicates proper usage of definite and indefinite articles. Likewise, one can further nuance the indication of proper usage by including additional permutations and tags to account for the specificities of languages other than English that the system will need to support. For instance, inanimate objects may be masculine, feminine, or gender-neutral in German. Consequently, to facilitate translation from English into German it is useful to explicate whether direct objects are animate or inanimate and gender-tag them, as in the example below:

```
"You {slay him[mn] | slay her[fn] | destroy the feminine-gender
object[f] | destroy the masculine-gender object[m] | destroy the
gender-neutral object}!"
```

Although superfluous in English, this gender-tagging feature is also useful in French, in which inanimate things are either masculine or feminine. In this example, two separate potential linguistic and translation issues are simultaneously resolved: first, the proper verb is chosen, so the game does not inform you that you have just "slayed" the barrel. In addition, the correct definite article is chosen based on gender. (In French, the correct form of the definite article depends on the gender of the corresponding noun).

In cases in which a paragraph or sentence string could give rise to a substantial number of possible permutations (which is rare enough, especially in English), the game chooses the metatag option that shares the most tag letters with the variable (negated letters counting as a match if they *aren't* present). In the case of a tie, the last tying match found is used. Here is a contrived example:

```
"As you kill {the[!n]} $NAME$, {it[i] | they[p] | he[mn] | she
[fn] | that named machine[n]} {screams | scream[p]} and {explodes
| explode[p]}!"
```

If `$NAME$` is "`Eric[mnv]`," then the string comes out "As you kill Eric, he screams and explodes!" This particular permutation is selected because "`he[mn]`" has two letters in common with "`Eric[mnv]`," which is more than any other choice. If `$NAME$` was "`Dr. Robot[ni]`," then the resulting sentence would be, "As you kill Dr. Robot, that named machine screams and explodes!" In this case, two different choices match: "`it[i]`" matches with one letter, and so does "`that named machine [n]`." Since there is a tie, the last valid match is chosen.

This brings us to the case of sentences that contain multiple "noun" variables. In such sentences, each `{}` must indicate the variable it is referencing. The variables are numbered, i.e., "`#1:$NAME$`," and the number used to identify a variable in the sentence is also used in blocks that correspond to that variable, such as a pronoun block, "`#1:{he[m] | she[f] | it}`." Consider this example of a complex string containing three variables as well as a personal pronoun, a direct object pronoun and a possessive adjective:

```
"#1:$PLAYER1$ gives #2:{the[!n]} #2:$ITEM$ to #3:$PLAYER2$.
#3:{He[m] | She[f] | It} thanks #1:{him[m] | her[f] | it} and goes
about #3:{his[m] | her[f] | its} business."
```

One possible rendering of this string would be, "Bob gives the sandwich to Sue. She thanks him and goes about her business." Or, if Sue gives the sandwich to Bob, the gender of all the pronouns and the possessive adjective is reversed.

There is no limit to the potential complexity of linguistic issues that one could address using a meta-language, but practicality wins out after a few days of coding. It simply is not worth spending a day of programming time to support additional verb tenses or language constructs that will not be used in the game. The best strategy is to start with a simple meta-language such as the one presented above, and to set aside a day or two of coding time for adding additional features when they come up. Every game will need a little bit of special code for its particular sentence structures.

In *Asheron's Call 2*, we extended the meta-language to support our randomly-generated treasure, which took the form "`$ADJECTIVE$ $NOUN$ $PREP-PHRASE$`," such as "Deadly Sword of the Sea." Because of the adjective, these items needed a little extra help from the meta-language to parse correctly in our various target languages. This wasn't a generic meta-language feature; other games wouldn't need to support adjectives in this way because most games don't create random items like this. Let's look at a few other handy features that are "generic" enough to add to most any game.

Sometimes string authors may wish to refer to an object without even displaying the name of that object at all (such as when we want to refer to something by pronoun only). Easy enough: if a variable is given a negative number, it does not appear in the string, but it can still be used by meta-cases:

```
"#-1:$FIRSTPLAYER$ The withered old man shakes his head sadly and
says, 'I already heard about it from #-1:{him[m] | her[f]| it}.'"
```

Another feature useful to a meta-language is automatically combining whitespace. For example, if given the following string:

```
"You    cannot    do    that!"
```

The game will display the sentence as follows:

```
"You cannot do that!"
```

Whenever more than one space appears in the final sentence, the extra spaces are automatically removed. This is useful for allowing extra spaces in between blocks of meta-language to improve legibility. Thus the designers can author a string as follows:

```
"When you pick up {the [!n]} $ITEM$, { he [m] | she [f] | it }
explodes in your hand!"
```

Although they included the extra spaces, the designers need not worry that the displayed sentence will contain two spaces after "he." They can put extra spaces wherever they want and the string will present correctly without any extraneous spaces. In *Asheron's Call 2*, this feature worked very well for English, but wasn't as useful for other languages. The German translators, for instance, often wanted the blocks to combine without any spaces at all. As a result, the German meta-strings tend to be a little harder to read than the English strings because they don't have any space in the expressions anywhere. Perhaps a future version of the meta-language could include additional whitespace-oriented features as well as the ability to configure them on a per-language basis.

The meta-language also assists in dealing with literal numbers. In many instances, variables in a sentence are replaced by numbers whose precise value is determined as the game is played. Consider items that are acquired or used up during play:

```
"You have $NUM$ arrows in your quiver."
```

Likewise, game designers will often wish to modify the wording of a sentence to distinguish between a singular and plural number. In the example string above, if the variable were singular, they would want to replace it with "1 arrow," but if the variable were plural, they would insert "5 arrows," for example. In *Asheron's Call 2*, we addressed this issue by adding a meta-tag, `[1]`, that is automatically applied by the meta-language system, instead of being applied by the string's authors, as other meta-tags are. When a variable is a singular number, such as "1," the game automatically assigns it the `[1]` tag, so it becomes "`1[1]`." Then we can include the singular and plural cases in the same sentence:

```
"You have $NUM$ {arrow[1] | arrows} in your quiver."
```

In English, only the number "1" receives the `[1]` tag. But that isn't the case for other languages — in French, the number "0" is also tagged with `[1]` because the French language treats both 0 and 1 as singular. And if we were translating into, say, Serbian, which uses multiple plural forms depending on the exact value of the number, the game would have to apply different tags depending on the exact value of the number. In some languages, implementing this feature may prove very difficult. This is the danger of adding features to a meta-language — if the game designers aren't fluent in all the target languages, adding new features can be risky and error-prone. It is always wise to show the proposed meta-language to the translators who will be using it, before coding the meta-language. Of course, translators will need to know what languages the meta-language is trying to support before being able to validate it; and determining which languages to support can be a problem in the games industry. Often, games are not developed with localization in mind, and when a developer requests clarification from the publisher as to which languages should be supported during development, the publisher won't know — that decision is typically made late in the development of the game.

Localization and foreign-language support is an issue that needs to be resolved with the game's publisher as early as possible. Meta-languages are most useful if they are developed early in the game, so that the designers can author strings using the meta-language immediately. If the meta-language is added later, someone will have to go back through all the text of the game and retrofit the meta-language onto the existing strings. Sometimes a good compromise is to get the publishers to commit to a list of the *possible* languages. Even if the game is ultimately not translated into all those languages, you can nonetheless create a meta-language that will support them and be in much better shape to support the languages into which the game is in fact translated.

The meta-language presented here supports many languages reasonably well, including the most common Asian and European languages. But not all languages can take advantage of this style of meta-language. The main limitation is that the plugged-in nouns have a fixed spelling — the meta-language can't change the nouns that get inserted into the sentence as variables. (Instead, the rest of the sentence changes based on the spelling, gender, and number of the noun.) For example, consider this string:

```
"Hey, look at this $ITEM$!"
```

The meta-language can change the string any way it wants, except that it can't actually change the spelling of the variable, `$ITEM$`. It can tack suffixes onto the end of `$ITEM$`, and this handles many common cases, but there is no way to actually change the value of `$ITEM$` itself through the meta-language. As a result, languages such as Inuktitut, which rely extensively on changing noun forms, are very poorly handled by this meta-language. To support these languages, the meta-language would have to be improved to allow the transformation of the plugged-in variable.

Even in languages that are well-supported by the meta-language, there will always be strings and scenarios that will not allow for grammatically flawless translation. Since meta-languages are extreme simplifications of complex human languages, there will inevitably be instances in which they will show their limitations. In some cases, the source-language string may need to be re-coded to better facilitate accurate translation into the target languages. A meta-language isn't a replacement for clever translators and conscientious designers — it is merely another tool to help achieve good translations.

Even though it clearly enhances translation quality, the complexity of the meta-language raises the possibility that the translation team may balk at the idea of using the meta-language at all. Though most translators are eager to use methods that improve translation quality, a meta-language can still be intimidating, especially to less technically-minded translation houses. And a meta-language certainly increases the complexity of translation. The game designers only need to write the strings once, but the translators will need to translate those meta-sentences into every other supported language. This will definitely increase the translation time, requiring translators to work longer hours to maintain their daily output and complete the project on time. Translators might pass the cost of this extra work on to the developers, either by raising their rates per translated word, or by counting meta-letters as additional "words" for purposes of billing. Some game projects might simply not be able to afford the additional cost. However, when developers are weighing the cost of using a meta-language, it is important to keep in mind that it will also save time, as it adds contextual information to the original strings. By using the meta-text as extra clues, translators are more likely to be able to figure out the function and context of the different strings in the game. As a result, fewer queries will be submitted to the designers, in turn yielding a faster turn around.

In the end, the decision as to whether to use a meta-language for game localization will depend on the project, the budget, and the translation team. But even when the translated versions of the game don't use a meta-language, it still can be used in English (or whatever native language the game is written in), and then be automatically removed from the strings sent for translation. To remove the meta-language, one need only write a script that parses every string in the game, and converts each instance of the meta-language into the appropriate article, pronoun, possessive adjective, etc., based on the value of the last tag in each instance, which, as described earlier, is the "default" case:

```
"You take {the[!n]} $ITEM$ and store {him[m] | her[f] | it} away."
```

Becomes:

```
"You take the $ITEM$ and store it away."
```

One could then send these simplified strings to the translators, who will still be confronted by the traditional problems of pronouns, gender, etc., but they won't be any worse off than they would have been if the game had not used a meta-language at all, and the English sentences will be of higher quality than would have otherwise been the case.

However, if the meta-language is to be used only for the native language, and not for translated versions, it cannot be used for too much "fancy stuff," or the translators will be unable to do their job effectively. The best practice in this case is to limit the use of the meta-language to articles and pronouns, such as "a," "an," "the," "he," or "she." It is definitely not advisable to use the meta-language to insert several different verbs or adjectives into one string. In any instances where the meta-language would otherwise be used to insert verbs or adjectives in this way, it is necessary to create a separate string for each verb or adjective. For instance, the following example would cause problems during translation:

```
"You meet $PERSON$ and {kiss her hand[f] | shake his hand[m]}."
```

In this instance, the meta-language is being used for more than just articles and pronouns: in fact, the entire meaning of the sentence changes depending on the gender of the person. If the translators aren't using the meta-language, their version of this string could only convey one of the two possible meanings, unless they combined the two possibilities into an unwieldy and grammatically incorrect hybrid. This string would need to be replaced by two separate strings, one for males and one for females.

Conclusion

The biggest fear Turbine Entertainment had during development of the meta-language was that it would greatly increase the time needed to author strings. Actually, though, only a small number of strings contain complex meta-language. Big blocks of NPC dialog, description, and exposition tend to contain absolutely no meta-language because they are not interactive — those strings don't contain any variables. In the end, the meta-language incurs a fairly small overhead for a savvy game development team.

Of course, any cost must be weighed against the associated benefits. In this case, the payoff is better grammar, both in the native version of the game and in translated versions. The importance of high-quality sentences seems subjective, but poor grammar can directly affect sales and player retention figures — incorrect grammar looks "buggy" to players and game reviewers. Certainly a meta-language is not practical for all games, but as more and more MMORPGs appear on the market, they start to differentiate themselves by their quality as much as by their features — and by the sheer amount of content they have, in the form of text. And the more text a game has, the more beneficial a meta-language is.

Because an MMORPG is a living game — one that continuously evolves, even after it ships — the traditional game model of "code it first, localize it later" doesn't work very well. The localization of MMORPGs yield far better results when proactive approaches are taken. As we've seen, even the native version of the game will improve as a result.

PART 4

Terminology management

A practical case for managing source-language terminology

Robin Lombard

Introduction

Any translator who has ever done any translation work for a software company may be familiar with that frustrating feeling of trying to translate a term that he or she does not fully understand: the glossary provided does not contain this term, and queries to the point of contact at the software company have yielded precious little information. What's more, the translator thinks she or he has seen another term that may be a synonym, but the software company contact can't provide any definitive information about this term, either. Don't these software companies understand how important it is to document and manage terminology systematically?

The benefits of managing localized software terminology have been well documented in recent years. In fact, localization is often cited as the main reason for managing such terminology (Corbolante and Irmler 2001; Hofmann and Mehnert 2000; Jaekel 2000; Warburton 2001). While it is true that a modest number of software companies have successfully created comprehensive processes and infrastructure for building internal multilingual termbases (IBM and SAP, for example), such companies are the exceptions rather than the rule. These companies practice source-language terminology management for software, which means that they collect and document terminology in the language in which the software was created. But managing source-language terminology is the key to more than consistent, accurate localization; it also plays a major role in customer satisfaction, usability, and trustworthiness in all languages.

Despite the increasing imperative for terminology management in government agencies and global businesses in general (Pavel and Nolet 2001; Wright 2001), current practices suggest that source-language terminology management remains a rare occurrence in the software industry, especially among those who are not involved in the localization of a product. This article focuses on source language terminology management at software companies: after defining source terminology, it analyzes the reasons why many companies don't do it, describes the benefits of managing source-language terminology, and offers some recommendations for making a business case in favor of source-language terminology management.

Terminology management and unmanaged source terminology

In the *Handbook of Terminology Management* Wright and Budin define terminology management as "any deliberate manipulation of terminological information" (Wright and Budin 1997: 1). This is an admittedly broad definition, but the key word is "deliberate." Terminology management as a series of actions carried out in a planned manner ensures the availability of terms, definitions, metadata, and other information pertaining to terminology. Terminology management guarantees that terminology is a known entity. One manages terminology so that one knows what terminology one has. Unmanaged terminology, on the other hand, is terminology that has not been documented. It represents terms and definitions about which one knows little or nothing. Like other types of data, terminology is most useful when it is documented and organized.

At many software companies, "terminology management" follows a scenario similar to the following: After a source-language (typically U.S. English) product is created, a documentation specialist typically compiles a glossary (i.e., a list of terms and definitions in the source language). If the product is to be localized, the key source terms are collected and passed — along with the glossary — to target-language localizers. The "target language" is the language into which the source language is to be translated (Japanese, for example). Target-language localizers grapple with inconsistencies in the source terminology and may have time to query the software company about a few items. The timeline for providing localized versions is typically extremely short; even so, localized versions are sometimes more consistent than source-language versions, since localizers tend to pay much more attention to terminology and consistency than developers.

While the target-language terminology in this scenario qualifies as "managed" under the definition given in the *Handbook*, the source-language terminology does not. The fact is that most software companies have no specific process for managing terminology beyond collecting terms for their source glossary. Most do not, for example, check source terminology for internal consistency, or consistency across products. And many smaller companies may rely on the terms used by large software publishers, such as Microsoft, which themselves employ inconsistent terminology.

Traditionally, the localization community in each company by default has been most likely to try to convince source development teams that terminology documentation is important. Even so, those who develop software, user interfaces, and user documentation may never see the localization costs of not managing terminology, ostensibly the most compelling reason to implement source terminology documentation processes. For example, a development team in the U.S. may not keep track of the terms it uses to reference a closing or stopped application, using multiple terms such as *cancel*, *quit*, *close*, *end*, and *stop* inconsistently in user interface elements and error messages. No one on the development team will think twice about this inconsistent usage — they know what these terms mean. What developers don't realize is that the localizer must treat every change in form as a change in meaning. If a localization vendor has a match for the term *close*, for example, but not *quit*, then she must do some research and determine if *quit* means the same thing as *close*, and whether she should then use the same translation for both. This probably involves contacting the vendor manager on the U.S. team, who must then try to find someone on the U.S. team who can definitively answer the question. It would not be unusual for five to seven people to be involved in the resolution of this kind of problem, each of whom spends ½ to 1 hour on the problem. This might cost a software company an average of $50 per hour per person involved, plus basic overhead costs. If the same question comes in from multiple languages to different U.S. team contacts, and the company maintains no central clearinghouse for localization issues, this could easily become a $1,000 question.

In addition, if the localization tasks are outsourced, the localization vendor may not (or may not be able to) push for increased efficiency on the source side. Thus the cycle of unmanaged terminology for all languages continues. This classic scenario has several repercussions: it enables the bad habits of those creating terminology in the source (in the U.S. this is typically U.S. English), it ensures customer confusion at some level, and it ensures that these software companies will continue to spend more on localization than they have to. The companies who put most of their energy towards cleaning up localization issues are in fact dealing with the symptoms and not the root causes of the problem.

A great illustration of the cost of not managing terminology is the error message, a dialog box that appears on the screen when a user or program has performed an unexpected or illegal action. A typical program can contain hundreds of error messages; larger programs, such as operating systems, contain thousands of error messages.

Let us consider a hypothetical example that illustrates how the failure to manage source terminology can directly impact costs associated with error messages. Suppose three developers working for the same software company each write an error message to the effect that a program has experienced a problem and must stop running. The developers work in different parts of the building and are working on different modules of code for the same program. Because no mechanism

- An error has occurred. Click **OK** to close this application.
- This function has encountered a severe error. Click **OK** to quit.
- Due to an internal error, this program must now terminate. Click **Details** to find out more about this error. Click **OK** to end this program.

Figure 1. Three error messages describing the same problem.

exists to standardize the terminology they use, each developer authors his or her own individual error message without stopping to consult with others, which leads to three different variations (see Figure 1).

Even though these three error messages describe the same problem, they are written differently and use different terms to describe the same concepts. Neither the localizer nor the user of the U.S. product will necessarily know that these three messages are reporting the same error. Similarly, in all likelihood they will not know that what follows "Click OK to" represents the same concept in each message. What started out as the same idea in three developer's heads becomes a source of cognitive dissonance to the typical user and a tangle of potential synonyms to the localizer.

Software companies typically calculate translation costs by the word, at an approximate average cost (across many different languages) of $0.25 per word. Table 1 provides an approximate translation cost breakdown for each error message given various conditions. It quickly becomes clear that the more languages a given piece of software is translated into, the larger the impact of unmanaged terminology. If, in this example, terminology were standardized by eliminating synonyms, costs would be reduced. For example, instead of using *error*, *internal error*, and *severe error*, all three messages could use *error*. By the same token, all three messages could use *program* instead of *function* and *application*, and *close* instead of *quit*, *end*, and *terminate*. By eliminating synonyms, a company could take advantage of the lower costs of recycling terms that have already been localized in a similar context. The table also indicates the relative translation costs that would be incurred if each of the three developers wrote 1,000 similar error messages, as well as the cost reduction that could be achieved by eliminating synonyms in those error messages in favor of standardized terms.

No matter how many error messages a company generates, it clearly spends more on translation if it does not manage their terminology. Of course, the best solution is to simply pick one error message for each possible program fault type and reuse it, saving potentially thousands of dollars in localization costs (not to mention the money saved by not writing and editing each different error message in duplicate, triplicate or worse).

Table 1. Error message localization savings achieved by terminology management.

Unmanaged source terminology	**Message 1**	**Message 2**	**Message 3**
Number of words	10	11	25
Number of synonyms	1	2	3
Cost of translation per language	$2.50	$2.75	$6.25
Cost of translation for 20 languages	$50	$55	$125
Cost per 1,000 similar error messages in an application	$50,000	$55,000	$125,000
Managed source terminology	**Message 1**	**Message 2**	**Message 3**
Number of words	9	9	22
Number of synonyms	0	0	0
Cost of translation per language if synonyms are eliminated	$2.25	$2.25	$5.50
Cost of translation for 20 languages	$45	$45	$110
Cost per 1,000 error messages in an application	$45,000	$45,000	$110,000
Savings achieved by managing source terminology	**Message 1**	**Message 2**	**Message 3**
Savings realized by replacing synonyms with standard terms	$5,000	$10,000	$15,000

What happens in the software development life cycle when no one is in charge of managing most or all of the source terminology and why is that such a terrible thing? Terminology can quickly become inconsistent within and across products, and source- and target-language glossaries can become filled with competing definitions. This can make products more difficult to use. For example, several years ago Microsoft designed and marketed two enterprise server products that were engineered to work together. However, the source terminology for the two products was developed separately. This resulted in competing term pairs such as *link state database* versus *link state table* and *container* versus *container object*. Even by comparing the source glossary definitions the non-expert could not determine whether these should be considered to be synonyms. Localizers probably had no choice but to consider them as separate concepts and translate accordingly.

If only a few terms in software products needed to be documented, perhaps unmanaged source terminology wouldn't be such a problem. But language plays a critical role in our ability to learn and master a computer program — if terms are not consistent then the program becomes harder for us to use. In addition, new technology and new concepts call for new terms. The terminology set for software will only become larger as programs become more ubiquitous and complex.

Why software companies don't manage source-language terminology

In my view, software companies don't manage their source terminology for at least three reasons: they don't understand its importance, they don't include it as a part of most product development cycles, workflow, and process, and they find it difficult to quantify return on investment.

Lack of awareness

Ideally, a software company undertakes source terminology management activities because it realizes that terminology is the key to customer satisfaction, usability, trustworthiness, and consistent localization. In most software companies, however, terminology is not a priority. As it was once explained to me, "terminology isn't sexy." Although software companies increasingly realize that products must be globalized, much of that focus is devoted to altering code so that the product functions in all locales and target languages; such projects tend not to emphasize consistent terminology (Brooks 2000; Thibodeau 2000). It is easier to calculate the cost of re-engineering code for local markets than the cost of dealing with unmanaged terminology.

A handful of software companies have succeeded in creating comprehensive processes and infrastructure for terminology management, including SAP (Childress 2002) and IBM (Warburton 2001; Granda and Warburton 2001). At J.D. Edwards, Ben Martin and Barbara Karsch calculated that it cost $2,000 to change a term in one language and only $150 to manage a term in one language (Martin and Karsch 2001). Thanks to compelling cost–benefit data such as this, they were able to implement a terminology database for the translation department at J.D. Edwards. Once this database was up and running, Karsch found that many different departments wanted to take advantage of it (Karsch 2003).

Despite the success of these few software companies, terminology management remains "under the radar" for most people in software development. No governing body for software terminology exists, so there is little impetus for standard industry terminology. It is arguably true that Microsoft and other large software companies are by default leaders in this area, and the standardization of Microsoft terminology could have a large trickle-down effect. But that alone has so far provided little incentive for those charged with bringing forth the next must-have technology. A similar problem has recently arisen in the field of nanotechnology: at the time of this writing, no standard naming convention has been adopted for systems of molecular size. Now those impacted (including insurance companies) have realized and pointed out this problem, so scientists have formed a standards group, and are in the process of creating standards and a "nano nomenclature" (Weiss 2004).

This issue tends to remain ignored in software companies because software developers are insiders to the program that they are creating — they can't think like the novice user because they aren't novice users. Many times I have been told, "My customer will understand this term, my customer won't understand that term" without any sort of usability data to back up the assertion. In most software companies, quality assurance doesn't include examining terminology for transparency. On the contrary, software developers carry on a tradition of creatively naming new concepts (Raymond 1993: 2). Raymond asserts that it is part of the so-called geek culture to use and promote language that distinguishes the insiders from the outsiders. Unfortunately, that perspective on terminology leads to just the sort of opaque terminology choices that should be avoided: those based on analogy and metaphor. Any broad usability study that includes novices and non-native speakers of English could show software developers and designers just how important easy-to-comprehend terminology is to their customers and the overall usability of their products.

Another reason for a lack of terminology management is that it requires a certain amount of centralization, whereas many software companies operate under a distributed model. For example, many large software companies develop multiple products simultaneously. In order to use terminology consistently across all products, they would have to maintain a central store and coordinate terminology decisions across product teams. If a company values distributed management and operations, a single-source and coordinated terminology effort may seem unnecessarily complicated and counter-intuitive.

It is important to note that not everyone on a user assistance or localization team may be aware of terminology issues. For example, why would anyone think that the same colors could be assigned different names in different products? Several years ago this happened with Microsoft Office products. Each product provided the user with a color palette that contained sixteen or more colors. Unfortunately, the user interface designers who assigned names to the colors in each product didn't communicate with each other. As a result, some Microsoft products ended up using different color names for the same RGB value (Brick Red and True Red, for example). Localizers pointed out the discrepancy, but it was too late (and too expensive) to change the products. Not only was localization money spent trying to fix the discrepancies in more than 20 target languages, extra money was spent initially in the design and authoring of the different color palettes.

Because there is no software terminology standards body, because developers and documentation writers often have no direct access to a customer feedback loop, because multiple new products are developed simultaneously, and maybe even because of company culture, those who have the power to name concepts might never directly feel the pain of poor or conflicting terminology choices.

Product cycle, workflow, and processes

In most software companies, an understanding of the terminology management problem is not reflected in the product cycle, workflow, and processes. Software companies, for example, treat their program code with extreme care. Random lines of code can't be checked into the code tree for a new application — there is a specific process that must be followed to test and verify that the code is going to work. Terminology in the software industry, on the other hand, is often created without following any term creation rules in the source-language (often resulting in opaque terms based on analogy and metaphor) and in many companies is seldom checked for consistency. Terminology is simply often not considered important. (I've often heard the excuse that terminology is "just words.")

Where in the product cycle is terminology typically collected and rationalized (i.e., checked for consistency and the presence of synonyms) against existing terms? Very few software companies follow this type of process as a formal part of the product cycle. Sue Ellen Wright has investigated this issue in industry and sees terminology management as an activity that should begin during the design phase (Wright and Budin 2001: 876). For some software companies, terminology management is synonymous with localization. Localization work begins as early as after the first beta release of the product, which means that terminology work starts at this point. However, this is too late for source language terminology management. Once a name has been conferred on a feature or technology by a developer, it often is then repeated in the specs and included in the user interface. By the time the terminology collection (or localization) step in the product cycle rolls around, there is too much investment in the original term provided by the developer to make a change.

What workflow does terminology management follow? In most software companies terminology management workflow exists for two particular categories of terms: terminology important to the marketing of a product (important features or technologies) and terminology that has created localization issues. The terminology related to marketing is usually the best managed terminology. In my experience, marketing employees will discuss early in the product life cycle what feature names will resonate most with customers, and may even do some kind of customer surveys on this terminology. They will also have access to legal research assistance to ensure there is no infringement on copyright or trademarks of other companies.

Localization workflow for terminology typically follows a cumbersome path. The localizer (whether in-house or outside the company) will send a terminology question to a designated contact in the company localization department, user assistance group, or release management. Most often, this contact will be unable to provide a specific answer for the terminology question, and will instead need to check with several other people. Sometimes the contact will be able to get back to the localizer with a satisfactory answer; many times there is no true answer, only the confirmation that indeed some sort of ambiguity exists.

Because it tends to be an afterthought for all but the most visible and marketable features, source terminology is typically not developed under any sort of tightly-controlled process or workflow. Even though most software companies don't deliberately manage source terminology, most have processes for collecting terminology at the end of the product development cycle and including part of it in user glossaries, product documentation indexes, and online Help. But the focus of these processes is limited to a single product — even if inconsistencies in terminology were spotted across products at this point, it would be too late to change anything.

Return on investment (ROI)

The most daunting problem, of course, in convincing any software company of the benefits of managing terminology is how to quantify the return on investment of terminology management. Sue Ellen Wright has laid a foundation for any company that wants to calculate the cost of terminology management (Wright 1997). However, such calculations may prove to be problematic. Wright points out that managing terminology generates only fixed costs, whereas not managing terminology generates only variable costs. She shows that the variable costs, along with the time wasted by employees trying to research and resolve conflicts, are likely greater than the known quantity of the cost of setting up a terminology management system. This is especially true in larger companies, who have to control larger terminology sets.

Nevertheless, quantifying ROI requires that one demonstrate how much the company will save and how quickly it will realize the savings. The ROI of terminology management is difficult to calculate because while one can easily determine how much it will cost to implement the new approach (fixed costs), one is still hard pressed to accurately calculate the scope and cost of current practices (variable costs). In terms of fixed costs, a company must determine (1) how much it costs to buy or set up a relational database to track source terminology, as well as (2) how much it will cost to populate and maintain such a database. In terms of variable costs, a company must determine the total amount currently spent (1) dealing with terminology issues, (2) performing and managing multiple cases of research, writing and editing of glossary items, (3) fixing terminology-related software "bugs," and (4) addressing complaints about lack of consistency and usability. A company must also determine the impact consistent terminology will have on ease of use of the software. What is that worth to a company? Sometimes trustworthiness is worth more than a lower price versus the competition. In the end, it may not be possible to determine exactly how much a standard, single source of terminology will save the company.

How can one show return on investment for terminology management? This is a hot topic among software companies as evidenced by the recent Localization Industry Standards Association (LISA) questionnaire and subsequent publication that analyzes the costs of terminology management based on a survey of eight software companies (http://www.lisa.org/products/reports/termreport2003.html). Unfortunately, there is no easy answer to the problem of ROI. Software companies may obtain different rates of ROI based on the number of terms they want to document, the number of products they have, how well terminology management activities fit into their existing workflow and processes, and the number of languages they localize into (the more languages, the higher the ROI of managed source terminology).

Ultimately, each business must obtain metrics that are most applicable to its particular situation. Return on investment figures, of course, can help convince those who make budgetary decisions that terminology management is a wise investment. Companies that have successfully implemented comprehensive, multilingual terminology management systems have had upper-level management support for their terminology management efforts.

Despite the work to date in ROI for source-language terminology management, there is no simple formula that companies can use to quickly determine how much they will save. In addition, showing a return on investment may take a few years, depending on the size of the company, volume of terminology, and other factors.

Benefits of managing source-language terminology

The benefits of managing source terminology in the software industry are far-reaching. Aside from any monetary savings, when a company documents its terminology it can ensure a better quality localized product, provide a less intimidating and easier-to-use environment for customers, and enhance the company's reputation for trustworthiness. By actively managing and standardizing their own terminology, larger software companies may be able to develop and coordinate cross-industry standard terminology, which would benefit the industry as a whole.

Managed source terminology ensures easier (and therefore less costly) software localization. It also contributes to a more user-friendly localized product, which may have an effect on how that product is perceived in a local market. It is important to multiply the impact of clear and standard source terminology by the number of languages that a software company supports. By ensuring standard, clear source terminology a company may be impacting products in 20, 30, 50 or more languages. Note that a software company cannot immediately ship localized versions ("sim ship" versions) of high quality and refuse to manage source terminology. Also, in order to take full advantage of the next wave in localization, machine translation, a company must maintain documented (and consistent) source-language terminology.

When a company manages its source terminology, it is easier for customers to use that company's software, because they do not run into cognitive barriers posed by multiple synonyms and cases where the same term represents different or overlapping concepts. In the course of my volunteer work as a computer tutor at the North Seattle Family Center, I have had several interactions in which people asked me the difference between "close" and "quit." After researching this issue, I discovered that in most cases there is no specific reason why both of these terms are used. In most contexts they do mean the same thing. However, in some applications these terms can mean different things (stop and do not save changes versus stop and save changes). But this distinction is so fine as to be of no use to the majority of users. By using these two terms where one would do, a company (in this case Microsoft) makes the program appear to be more complex than it really is. If these terms had been caught early on and standardized within and across products, the user would not be confused.

After a company begins to manage its source-language terminology, it becomes possible for that company to provide a single source of those terms (including definitions). Maintaining a single source of terms and definitions allows a company to reuse terms and definitions across products, ensuring consistent glossaries. A single source also provides a common point of reference for any employee working on sales, marketing, development, or documentation materials. Suddenly, the company has the potential to speak with one voice. This is a big benefit for customers, who will not be required to learn so many new terms with each new product. The customer also benefits if the company can publish a single, consistent source of terms and definitions for those both inside and outside the company to use (for example, see the IBM glossary available on the company's Web site at http://www-306.ibm.com/ibm/terminology/).

A company that uses clear, consistent terminology is more likely to be trusted by customers and partners. If customers see that a company is consistent on the outside, they are more likely to believe that company is detail-oriented on the inside. Consistent terminology also demonstrates a commitment to and focus on the user. Trustworthiness is especially important today given the widespread dissatisfaction with spam, pop-up windows, viruses, and other security- and privacy-

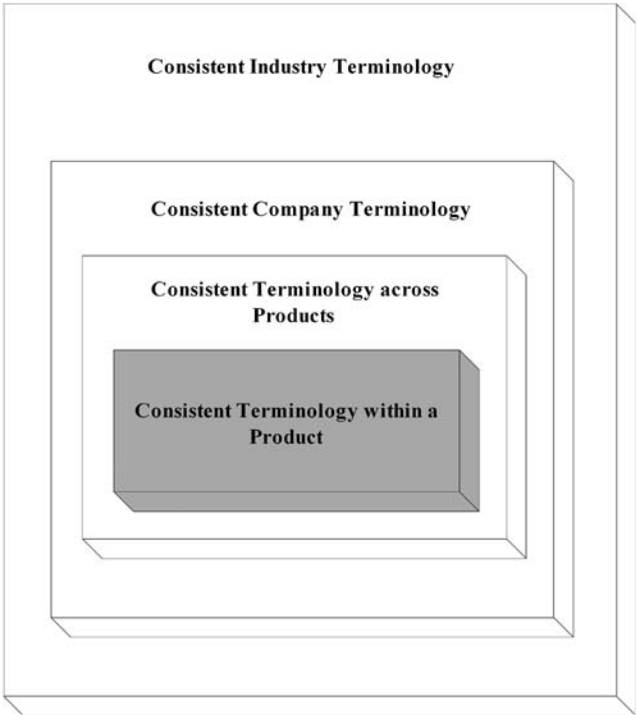

Figure 2. Potential cumulative effects of terminology management.

related threats. Anything that a company can do to support its trustworthy reputation is important.

Figure 2 shows the cumulative effects of source-language terminology management in the software industry. By creating a cross-company standard terminology set, a software company is taking the first step toward being able to participate in cross-industry terminology standardization. It is also a step towards creating a more user-friendly and trustworthy industry.

The benefit of terminology standardization across the software industry would extend to developers and IT professionals who develop third-party software or software in mixed operating system and application environments. Terminology standardization would allow these software creators to more quickly and easily create products that are easily used and localized. Currently this is not the case. For example, if Microsoft's terminology is not standardized, and third-party vendors who are building applications to run on Microsoft software mimic Microsoft terminology choices, those third-party companies multiply the chaos. This issue is actually well known enough to be parodied on the Internet. The cartoon in Figure 3 illustrates the terminology dilemma of third-party software vendors quite eloquently.

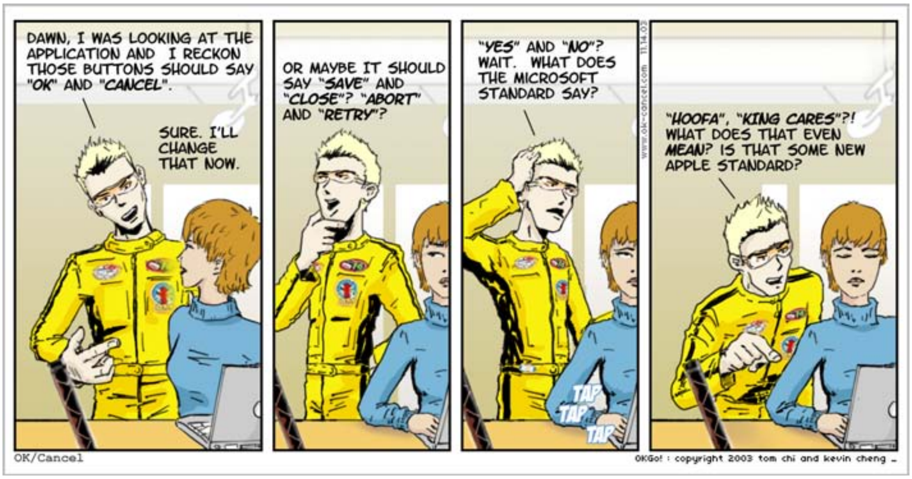

Figure 3. Software terminology parodied on the Internet.
© 2003 Tom Chi and Kevin Cheng.

A success story

One of the small success stories in which I have been involved at Microsoft is the standardization of the out-of-band release terms. In addition to regular product releases, Microsoft produces a variety of feature packs, updates, and other small code sets. These are referred to as *out-of-band* releases since they are not shipped as part of a regular product release. For years each product team had their own name for these releases: update, QFE, patch, hotfix, and so on. In the process of trying to standardize this terminology we found that not only did these terms refer to overlapping concepts across product groups, but they also implied different business processes.

It took several months of effort, but a small group at Microsoft came up with a set of standard terminology for the out-of-band release code sets. The terms and definitions in this set were also translated into approximately 26 languages and made available to localization vendors. But it wasn't enough just to document terms and definitions; we also had to work to make sure that employees knew about the new naming policy and implemented it. Though there was some initial resistance among employees to changing some longstanding terminology, there was support for consistent terminology in this domain from upper management. Eventually, most employees came around and surprisingly, there is now a sense of relief that this terminology is actually set as a company standard. Thanks to this standardization, employees no longer have to spend time researching the terms appropriate for their out-of-band release.

The benefits of this effort include clearer communication across teams about these releases, much more consistent localization, and an overall better customer experience.

Thoughts on building a case for managing source terminology

Building a case for managing source-language software terminology requires tackling at least five issues: a tool (based on a relational database), grassroots and upper-level management support, a limited domain, a set of metrics, and determination.

The role of the tool is to centralize the terminology. This is crucial. Before terminology standardization can be attempted, a company must establish a single location from which all employees access terminology. IBM, J.D. Edwards, and SAP AG (among others) have each created their own terminology management solution. This is costly and may not be a practical strategy for smaller companies. But there are several off-the-shelf terminology management applications for sale, and something as basic as an Excel spreadsheet will do in a pinch. The advantage of a spreadsheet program is that it is easy to use. I once managed 12,000 term entries in a set of Excel worksheets, but I don't recommend this method for more than several hundred terms as it quickly becomes unwieldy. Whatever method a company chooses, its terminology should be available from a central location.

But a terminology management tool alone is not enough. There must be some support among employees for a move towards managed source-language terminology.

My experience is that in all but the smallest software companies, one person devoting spare time to terminology management won't get too far. Successful terminology management requires support and assistance. Some say terminology management initiatives can't go far without upper-level management support, but in my experience you cannot get that support without first showing some progress and benefit. One way to show progress is to develop grassroots support. For example, if a localizer can find terminology advocates in other departments, together they may be able to drive small changes in process and workflow that will enable better management of terminology. With support, they may also be able to start a pilot project for a single product or domain in which they can show the value of terminology management. This is how the move to standardize source-language terminology began at Microsoft.

It is difficult for any software company that has never managed source-language terminology to begin doing so. This difficulty may be exacerbated if a centralized process such as terminology management is not naturally supported by the company culture. Part of the cost of setting up a terminology management solution may indeed be a change in paradigm, while part of the return on investment will be efficiencies that stretch to increased ease of use and less costly localization.

Metrics are critical to building a successful business case for terminology management. A company that wishes to track terminology management should begin by asking the following questions:

- How many duplicate definitions exist across the company for a specific concept? Each duplicate definition represents a duplication of research, writing, and editing tasks, and possibly duplicate localization costs in each target language.
- How many synonym pairs exist for a specific product? It is difficult to assign an exact cost to synonyms, but at the very least they will entail higher localization costs (in the form of higher word counts and time spent researching and untangling the synonym pairs in each supported language).
- How many terminology-related "bugs" are filed during a product cycle and on average how much time does it take to resolve them? Many companies already track this metric, and being able to show a substantial reduction here may be a good foundation in making a case for managing source terminology.
- In how many languages is the software published? The more languages a company needs to support, the more cost-effective it will be to invest in source-language terminology management.
- How much does it cost to localize each word? This cost will vary by localization vendor, language, and volume.
- How many fuzzy matches would become 100% matches if the source terminology were managed? During the localization process, 100% matches cost less.
- Are there any terminology issues that have ended up costing extra money because they had to be fixed at an expensive point late in the development process (after the product shipped, for example)? In addition to expense, mistakes arising from unmanaged terminology can result in legal action or simply embarrassment. Often, one or two such errors suffice to drive home the value of source terminology management.

Answers to these questions can provide the foundation for ROI measures. Each of the answers above reflects potential expenditures that a software company wouldn't have to make if it managed source-language terminology.

A company may need to start small and focus on just one domain or product that matters to upper-level management. After operating for 25 years without actively managing source-language terminology, for example, Microsoft began with just one group. We have since moved on to support other projects that are important to upper-level management, such as security, developing more robust processes, communication paths, and tools. Without upper-level management support, we might never have made headway toward a single set of terms and definitions.

Every company that has successfully implemented terminology management can testify to the fact that it requires a lot of dedication and determination. Setting up a database and convincing people what it costs *not* to manage terminology is only part of the battle. One must also change people's attitudes toward the importance of consistent, transparent terminology in software.

Conclusion

It won't be easy to reach a point at which source-language terminology management in the software industry is the rule rather than the exception. That vision can only be achieved if each company is committed to managing its own source-language terminology, and working with other companies in the industry towards standardization. Each company will need to determine the exact tradeoff between required investment and rewards, and how it can best integrate source-language terminology management into company workflow, processes, and culture. But the total benefit of such a commitment will far exceed the sum of each company's individual contribution: all target-language products and third-party add-on programs will also be able to leverage and pass on the more usable and trustworthy user experience enabled by consistent terminology.

References

Brooks, D. 2000. "What price globalization? Managing costs at Microsoft." In *Translating into Success*, R.C. Sprung (ed), 43–57. Amsterdam/Philadelphia: John Benjamins.

Childress, M. 2002. "Applying a knowledge management approach to terminology management." In *Proceedings of the 6th International Conference on Terminology and Knowledge Engineering*, 31–37. Nancy, France.

Corbolante, L. and Irmler, U. 2001. "Software terminology and localization." In *Handbook of Terminology Management, Vol. 2*, S.E. Wright and G. Budin (eds), 516–535. Amsterdam/Philadelphia: John Benjamins.

Granda, R. and Warburton, K. 2001. "Terminology management as data management." In *Proceedings of the 2001 Conference of the Centre for Advanced Studies on Collaborative Research*, D.A. Stewart and J.H. Johnson (eds). Toronto, Ontario, Canada: IBM Press.

Hofmann, C. and Mehnert, T. 2000. "Multilingual information management at Schneider Automation." In *Translating into Success*, R.C. Sprung (ed), 60–79. Amsterdam/Philadelphia: John Benjamins.

Jaekel, G. 2000. "Terminology management at Ericsson." In *Translating into Success*, R.C. Sprung (ed), 159–171. Amsterdam/Philadelphia: John Benjamins.

Karsch, B.I. 2003. "The Evolution of Version 2 — A multilingual database for a multitude of users." In *6th International TAMA Conference: Multilingual Knowledge and Technology Transfer*. G. de Schryver (ed), 105–11. Johannesburg: TermNet.

Martin, B. and Karsch, B.I. 2001. "Terminology management driving content management." Paper presented at the 5th TermNet Symposium: Sharing Terminological Knowledge. Antwerp.

Pavel, S. and Nolet, D. 2001. *Handbook of Terminology*. Hull, Quebec: Canada Translation Bureau.

Raymond, E.S. 1993. *The New Hacker's Dictionary, Second Edition*. Cambridge, Massachusetts: MIT Press.

Thibodeau, R.P. 2000. "Making a global product at MapInfo Corporation." In *Translating into Success*, R.C. Sprung (ed), 127–146. Amsterdam/Philadelphia: John Benjamins.

Warburton, K. 2001. "Globalization and terminology management." In *Handbook of Terminology Management, Vol. 2*, S.E. Wright and G. Budin (eds), 677–696. Amsterdam/Philadelphia: John Benjamins.

Weiss, R. 2004. "Language of science lags behind nanotech." *The Washington Post*. May 17, A7.

Wright, S.E. 1997. "Economic issues of terminology management." *TermNet News* (54/55): 1–10.

———. 2001. "Terminology as an organizational principle in CIM environments." In *Handbook of Terminology Management, Vol. 2*, S.E. Wright and G. Budin (eds), 467–479. Amsterdam/Philadelphia: John Benjamins.

Wright, S.E. and Budin, G. 1997. "Introduction." In *Handbook of Terminology Management, Vol. 1*, S.E. Wright and G. Budin (eds), 1–10. Amsterdam/Philadelphia: John Benjamins.

Wright, S.E. and Budin, G., (eds). 2001. *Handbook of Terminology Management.* Vol. 2, *Application-Oriented Terminology Management*. Amsterdam/Philadelphia: John Benjamins.

Terminology workflow in the localization process

Barbara Inge Karsch

Introduction and background

The importance of software localization has been steadily increasing ever since the late 1980s. At that time, the US software market began to saturate, and companies started to look abroad to tap into other sources of revenue. Today, international sales account for approximately 50% of the revenue of many major software products. Over the past 15 years or so, the advent and strong growth of international software sales have in turn led to an increasing demand for localization professionals. The release of a new product or major update by a software giant such as Microsoft may mean months' worth of work for many freelance translators from around the world. Furthermore, localization has evolved into a far more refined process than was previously the case. In the past, translation was an afterthought and translators would begin to tackle the entire project either after development had been completed, or at a point very late in the development process. Today, given the emphasis on developing and simultaneously shipping multiple language versions of a given software product, individual localizers often only focus on one discrete component of the overall localization workflow.

Terminology management has emerged as one such distinct task in the localization process. As a discipline, terminology management has existed for decades, even centuries: Oeser and Picht note that the roots of the discipline stretch back to the 18th century and identify Carl von Linné as the founder of terminology organization, standardization and planning. Terminology standardization in the realm of business is attributed to Johann Beckmann, while the theoretical foundations that govern terminology management today were laid in the twentieth century by Wüster, Drezen, etc. (Oeser and Picht 1998: 341f). Today, researchers and practitioners are integrating ideas from knowledge management and information science to improve their processes.

Since the advent of software localization in the late 1980s, terminology work has moved outside the realm of individual researchers and academic environments. Suddenly, there is a need to organize, standardize, plan and coin terminology on a large scale and for products that will sell and make a profit in dozens of countries. Indeed, in most cases today, an American software product is localized, at minimum, into Japanese and German as well as French and Spanish (Schmitz 2000: 1). But generally localization into a far greater number of languages is required.

The imperative to refine the terminology management process for large-scale, multi-language projects has also driven the development of skills and tools to facilitate the process. It is thus that the profession of terminologist has emerged. Many of the major translation programs are now offering at least courses, if not certificates and in some cases even degrees in terminology management.[1] The development of tools has followed a similar path of extremely rapid development, and in the space of a few short years the simple *ad hoc* spreadsheet with a few columns has given way to today's complex terminology management programs, which allow efficient management of and easy access to multilingual technical terms and idiomatic expressions (Schmitz 2000: 8).

Developing such a terminology management system (TMS) or buying an out-of-the-box solution can be rather expensive. It is even more expensive to maintain a terminology database. While the need for terminology management in the localization process is clear to translation professionals, it is constantly and justifiably being questioned by business managers. No business can afford to invest and develop a major component of a business process without proof of return on investment (ROI). Consequently, most terminology departments have had to analyze their process in order to justify the existence of their terminologists and tools.

Little information is available regarding the ROI of terminology management. One exception is the study that was conducted at J.D. Edwards in 1998. The results of this study indicate that in the J.D. Edwards environment, changing an unmanaged term in the translation memory for software and documentation cost approximately USD 2000 in *just one language* (Martin and Karsch 2001: 19). Once a terminology management system was in place, the number of changes declined and the consistency of terms used in the translated products increased. The cost associated with a managed term was about USD 100 in the beginning and dropped below USD 80 once the system was well established. ROI measures are hard to define, as many variables influence the localization process and terminology is just one aspect. It is all the more important that the process be as flawless and efficient as possible, so as not to incur undue cost.[2] Nevertheless, the J.D. Edwards data demonstrate that managing terminology at the source is orders of magnitude cheaper than fixing it downstream, after software and documents have been authored, and the greater the number of target languages, the greater the impact of source-language terminology management on cost savings.

The unit of information in a terminology management system is very small, but the typical system contains thousands of such units.[3] Moreover, each entry must contain the most valuable and up-to-date information. The system must be accessible by as many users as possible, and there must be an efficient way for these users to provide feedback. The entry must be complete and fit into the conceptual system. Finally, corrections must be implemented in a timely fashion, and most importantly all users must be looking at the same database. Given these requirements, without which ROI cannot be achieved, a Web-based system with automatic workflow that sends entries from one stakeholder to the next is the most efficient solution. The J.D. Edwards terminology database took just such an approach and was highly successful with it. After contextualizing terminology with respect to the corporate knowledge creation process, this article will describe the J.D. Edwards workflow. It will then analyze the natural workflow that a term entry goes through from inception through its indefinite life in a terminology database. The article will then describe the steps and their characteristics and modularize them for application in other settings. While this scenario presupposes an in-house translation environment, it is conceivable that external localization vendors could also participate in several of these workflow steps.

Terminology and knowledge

The product of the modern organization is no longer a tangible asset that rolls off a conveyor belt.[4] The main asset today is knowledge created by product contributors along the virtual conveyor belt, that is, the product workflow, which is influenced by a variety of factors such as the type of knowledge, the knowledge culture, and the knowledge infrastructure.

Types of knowledge

Ikujiro Nonaka (1998) identifies two types of knowledge: explicit knowledge that has been expressed in standards, specifications, etc., and which can easily be communicated and shared; and tacit knowledge that is highly personal and which is a function of an individual's know-how, beliefs and perspectives. In the localization process, explicit knowledge is captured in the style guide, automatic spell checker, terminology standards, etc. Tacit knowledge, on the other hand, is the unspoken operative knowledge drawn upon when a software developer devises a term for a new concept or when terminologists start the research for a particular term based on a hunch, for example. Tacit knowledge cannot be captured easily. But ideally, the application of tacit knowledge results in explicit information. It is important that this information be documented in a way that facilitates repeatability and reproducibility (see also Wright 2001).

Budin (2002) has broken down the knowledge that is required within the translation process into the following eight types: linguistic, translatory, subject-matter, information retrieval and organization, managerial, computer and other media, intercultural, and terminology knowledge. As a sub-process of the localization process, terminology management necessitates varying degrees of these explicit and implicit types of knowledge at different stages of the process.

Termbase entries are the explicit product of the application of these eight types of knowledge by skilled professionals. At the same time, they are themselves a form of knowledge. They must meet certain quality criteria that are set forth for each step in the workflow. Each and every terminological entry should be created with the expectation that it contain the best possible content available at any given point in the workflow. Even if not all data categories are filled in, the information present in the entry can be very reliable. The status field in combination with the workflow step should indicate to any user who queries an entry how reliable that entry is. Such information may also be combined and presented in the form of a reliability index. It should also be noted that an entry is only of value if it is embedded in its conceptual system. The first term of a conceptual system will certainly not fulfill this requirement. Nevertheless, the goal is to make explicit the conceptual system that lies beneath every software product. Explicit relationships between concepts, if not in graphical form then at least through cross-references,

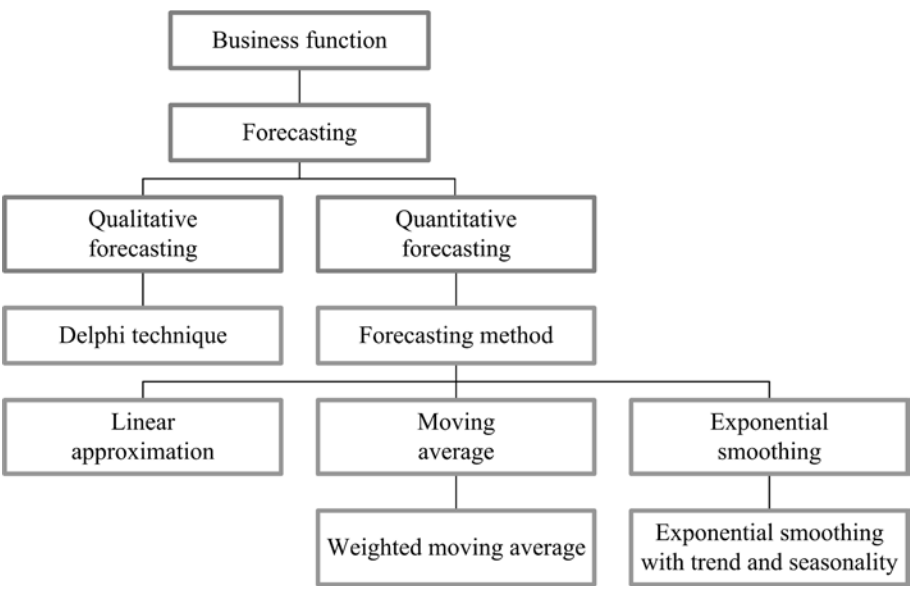

Figure 1. Example of a conceptual system (see also Karsch 2002).

accelerate the translation process. It is beneficial to relate, for instance, the concept "qualitative forecasting" to its coordinate "quantitative forecasting" (see Figure 1).

Knowledge culture

The culture of an organization exerts considerable influence over its ability to create knowledge. It goes without saying that willingness to share knowledge is a prerequisite for successful knowledge capturing. IBM's Tom Short (2004: 39) writes that "KM solutions rely heavily on the softer, human behavior and cultural aspects of business rather than on computer systems and technology." While most companies have recognized the need to share knowledge, the successful ones have also found ways to put knowledge sharing into practice. Peter Drucker (1998: 12) identifies four success factors, three of which will be mentioned here: a) reward systems, b) a unified vision, and c) an appropriate organizational structure.

In any organization in which knowledge hoarding is practiced and rewarded, a knowledge database will either remain empty or outdated, and therefore is doomed to be a useless and expensive tool. However, the vision of the knowledge-creating company sees terminology management as a problem that the organization must solve collectively, and not merely as a task that can or should be relegated to the individual translator. The solution to this "problem" requires a variety of skills, which are contributed by a virtual team of stakeholders. These experts require training at regular intervals and are rewarded for their work accordingly, but most of all they all share the same collective vision.

The J.D. Edwards model was successful largely because the culture of the documentation department supported terminology management. Firstly, the department changed from an extremely hierarchical and siloed production facility with poor communication flow between departments to a learning organization[5] in which translators and terminologists shared a common vision of the task and were rewarded for sharing and documenting what they knew (Karsch 2004). Over the years, the seven full-time terminologists worked with writers, editors, developers, marketing and sales personnel as well as translators. All participants in the workflow received a day of terminology training, which was offered upon demand; then they were evaluated and rewarded based on their contribution (measured in terms of both quantity and quality).

Another software company that has been managing terminology since the mid-80s is SAP. Technical writers and editors author definitions directly in SAPterm, three full-time terminologists review them, and translators attach foreign equivalents to the conceptual entries. The SAP model is successful because terminology management has been recognized as an integral part of software localization as well as information management and is receiving attention from product contributors on a consistent basis. Translators, writers and editors spend up to half a day per week managing their terms in response to a recommendation by upper management (Childress 2004a). Furthermore, training is offered in many SAP locations all over the world.

Knowledge infrastructure

Successful knowledge management tools originate from the analysis of the underlying business process. In most in-house environments, terminology is being created, reviewed, localized and updated either in waves or on an ongoing basis. At the same time, terminological entries represent very small units of information. Therefore, the knowledge infrastructure should allow for easy access and transfer of information whereby entry contributors can quickly and efficiently create, disseminate, and update entries. Knowledge consumers, on the other hand, should be able to query, retrieve and critique an entry in an easy manner. Efficient feedback mechanisms must be in place so that knowledge from around the company can be gathered and made available, every time an employee looks at an entry. This ensures that each person involved as consumer or producer of knowledge contributes his or her best skills and that each step adds value.

It is much easier for users to grasp the content of an entry if all entries are structured similarly. For this reason a standards document must be in place to guide contributors in filling in individual data categories in a consistent fashion. This is not to say that different entry types must cover all of the same data categories, but it means that any information entered should conform to the documented norm.

An automated workflow system is the most effective way to enable work on terminological entries by numerous stakeholders that have different, yet essential skills. Creating and storing a term in a single database enables all entry contributors to work on it, and all consumers to query the term and follow its development. The database may also allow users to track the progress and reliability of the entry via its status and, in more sophisticated systems, via its reliability index. Today, such a system can easily be made accessible to users around the world. This Web-based infrastructure translates into an even richer knowledge resource. The key is that everyone uses the same database and can react to changes immediately. In addition, context-sensitive help can be built into the system to indicate the description of the data category field following the ISO 12620 definitions (ISO TC37/SC 3 1999). This will ensure that, for instance, someone who is requesting addition of a new term and who may not be proficient in terminology management has help at his fingertips.

Childress describes how server problems at SAP and the lack of user-friendliness of the tool made terminology work difficult at one point in the development of the proprietary TMS. The immediate consequence was that writers, editors, and translators all stopped using the tool. One of the first decisions was that there be only one data repository, and where necessary, this repository should be linked directly to other language tools (Childress 2004a).

Bowne Global Solutions,[6] a provider of localization services, also developed a very sophisticated proprietary tool. According to Daniela Dettmann (2002), customers as well as localizers anywhere in the world access the live term database in TermGlobal™ to update terminological entries in real time. This assures that localizers working on the same project always have up-to-date information on hand.

The J.D. Edwards terminology workflow

The J.D. Edwards terminology management system (TDB, for "terminology database") provides an excellent case study for just such a workflow tool. J.D. Edwards, a software company that was bought out by a competitor in 2003, developed enterprise software with a focus on financial, manufacturing, and distribution processes, among others. Software developed by J.D. Edwards was translated into 21 languages by an in-house staff of 80 translators and terminologists. Every translator and terminologist specialized in at least one of the business areas, called verticals (e.g., Distribution, Financials, Human Resources, or Oil and Gas). The in-house tools and terminology team developed the terminology management software that will be described below.

Technical specifications

TDB Version 1 was a client-server software application written in Microsoft® Visual Basic and installed on an SQL Server database. To make content accessible to users as far away as China or Brazil and to provide the capabilities required by the J.D. Edwards Marketing department, a Web-based version went live in late spring of 2003. The program comprised 50 database tables, version control, role-based security, and the basic functionality to provide one-dimensional conceptual systems. The "Search and Request," "Research and Validation," "Release," and "Foreign Entry" steps of the integrated workflow depicted in Figure 2 will be discussed in the following sections.

Search and request

Searches are carried out to meet a variety of needs: a translator may be looking for a concept; a writer may be investigating usage; a consultant may be searching for

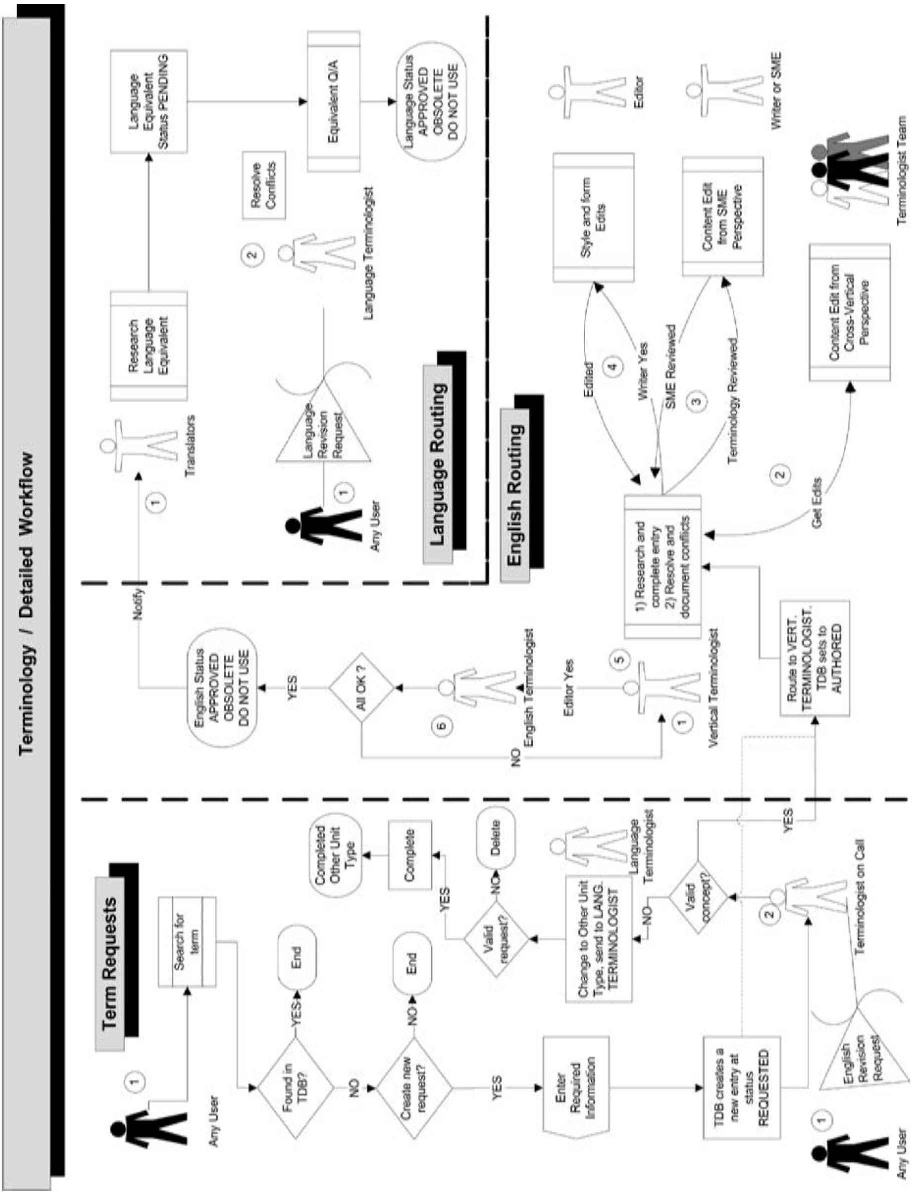

Figure 2. The J.D. Edwards terminology workflow (see also Karsch 2002: 175).

the translation of a particular term; or a terminologist may be doing systematic terminology work. Every time a concept or term is found, the search ends and the database has served its purpose.

In the event that a search yields no results because a term is not present in the database, the searching party can trigger the launch of a request form. To do so, the originator must provide all required information: J.D. Edwards' standards mandate that the term, context, part of speech, and vertical all be supplied. The entry form also contains a Comments field that allows for additional information and communication between the requester and the researcher. Upon submission, the entry's status is automatically set to "Requested."

Once the request is triggered, it is forwarded to the inbox of the Terminologist on Call (TOC). The TOC checks the request for validity, i.e., researches whether the concept already exists in the database under a different name, checks spelling variations, and ensures that all mandatory information is filled in. The entry is then assigned to the vertical terminologist and its status is set to "Work in Progress."

Research and validation

The vertical terminologist researches the entry, writes a definition and presents the entry to the terminologist team. All terminologists validate the concept against existing concepts in their own vertical. Most synonyms are discovered during this step. Furthermore, terminologists check whether standards have been adhered to. Any feedback is evaluated and incorporated before the entry is moved on to a subject matter expert (SME), who validates the content of an entry against the subject field. Any suggestions for improvement are entered into the Comments field and the entry is sent back to the vertical terminologist, who applies comments as warranted. The next step is linguistic validation by an editor. Again, feedback is entered either directly in the data category fields or elaborated on in the Comments field. The vertical terminologist enters feedback and makes sure that all pending issues are resolved, and then forwards the term to the English terminologist for release.

Release

Although a "requested" or "work in progress" entry is visible in the database to anyone who searches for it, it is not released, i.e., approved, rejected, or obsoleted, until the English terminologist validates the entire entry for data integrity. Specifically, the entry must be validated against a checklist of ten quality-related items before it can receive its final status of "Approved," "Obsolete" or "Do Not Use." The use of this checklist ensures that the entry is as good as possible at that particular point in time.

Foreign entry

Although language terminologists and translators may be working on a particular foreign language entry while the corresponding English entry is still in process, most of the work should be done on the foreign term only when the English entry is fairly stable or even released to avoid the risk of major rework. A foreign entry is typically researched and checked by at least two people, the terminologist and the translator. In many cases, the entry is also presented to a consultant in the foreign subsidiary for additional content approval. Once all parties have agreed and all issues are resolved, the foreign entry is released.

The individual pieces of a terminology workflow model

While the above case study is a good model for one possible deployment, it is also necessary to examine each step in the workflow from a knowledge management perspective. In terms of knowledge management, the first step in the process is the conception of an idea which is then formalized and input into the system. This input is checked against various standards. After standardization has been confirmed, the overall quality of the entry must be checked again, after which the entry can be released. The output and use of the database can take various forms. The last step, i.e., ongoing review via feedback by users of the database, can overlap with the previous steps, but will continue for as long as the entry is maintained in the database. Figure 3 illustrates these steps in relation to adherence to standards and quality as well as time and exposure to users.

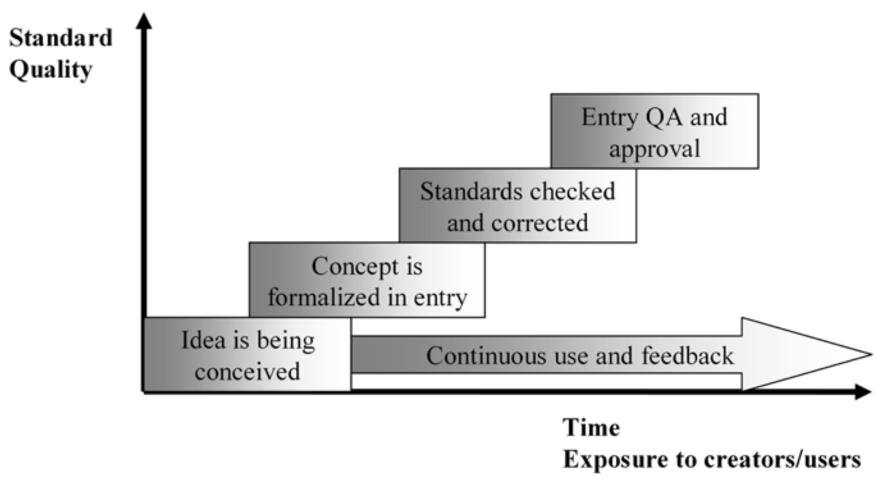

Figure 3. The life of a conceptual entry in relation to quality.

Conceptual input

As Nonaka has noted, "New knowledge always begins with the individual" (1998: 26). Individuals in various departments in an organization create new conceptual or terminological knowledge, which represents the main input of a terminology workflow system. In an ideal scenario, the primary conceptual input comes from the product team (e.g., product manager, developer, tester, primary writer and/or editor) in charge of the functions and features of a particular product. This is not to say that the conceptual input or ideas are necessarily conceived by this team. Since functionality should be customer-driven, a particular feature/concept and its name may well be the suggestion of a customer. In other cases, a feature might reflect standards that were produced by a standardizing body, for example, accounting functionality regulated by the Financial Accounting Standards Board. Nevertheless, since the product unit gathers all such information, it shall be considered here as the primary source of conceptual input. This is the ideal case, in which the process is proactive and terminology is managed systematically.

On the other hand, the process is both *ad hoc* and reactive when a concept is not documented immediately after it enters the conceptual world of a company. In such cases, individuals downstream in the product creation process may request information about a concept. These secondary contributors to the terminology workflow may be other writers and editors, translators and terminologists, dependant product units or foreign offices. In the reactive workflow, such contributors issue term entry requests and supply as input what they have found in a particular context. It is even conceivable that tertiary contributors such as customers, software developers of external, yet dependant products, etc., may request information about a certain concept or several concepts that have not been documented. They, too, may contribute by indicating the context in which the concept in question occurs, but more than likely, they will not be working in the TMS.

Existing knowledge management systems are often criticized for producing overly rational, static, and non-contextual information. Malhotra states that "the prevailing knowledge management paradigm limits itself by its emphasis on convergence and consensus-oriented processing of information" (Malhotra 2000: 37). Since much of this terminology workflow relies on computers and computer programs, it is important that the system input be a product of the very essence of being human, i.e., the capacity to interpret and make meaning of information. Furthermore, as mentioned earlier, it is not easy to explicate tacit knowledge. In the words of philosopher Michael Polanyi, "We can know more than we can tell" (Nonaka 1998: 27). If standardization, whose important role in the localization process will be discussed later, is imposed too soon in the process, the conceptual input will clearly suffer.

Nonetheless, it is important that primary contributors receive basic terminology training. For example, they should understand what a concept-driven system is, and they should know what synonyms are and how to search for them. Applying this type of knowledge in the first step of the workflow will prevent duplicate entries — one of the main challenges of terminology management.

In the terminology process, creativity and meaning-making occur during the naming of concepts, in both source and target languages, as well as during definition authoring. Naming the features and functionality of a software program requires great skill. Due to the space limitations and the lack of contextual material on a software screen, it is of utmost importance that features be named concisely and aptly. The resulting denominator is in many cases a neologism that will either be adopted or rejected by the language community. It is advisable when creating concept denominators to engage in a collective effort in which the responsibility for coining a phrase is not left to an isolated developer or technical writer from a product unit, but rather in which suggestions are gathered, debated and adopted or rejected. During this process, the tacit product knowledge of stakeholders is tapped and documented. It is crucial that this knowledge capture occur during the inception of a new feature, since participating stakeholders may be shifting into different roles, leaving the organization, or might simply not remember their original reasoning.

A live example of this process, albeit for target terminology, is the Microsoft LIP project in which selected members of a linguistic community vote on or suggest translations for particular software features (Microsoft New Zealand 2004).

The product of this knowledge explicitation process is documented in the terminology system, wherein the name of a particular feature is attached to its description. This description is usually a definition that can take various forms. Here, too, it is important to allow for a certain freedom of expression. While a terminological definition must follow certain standards, the initial draft of the entry may look very different from the final form. The first draft or drafts are products of research as well as brainstorming. A skilled technical writer, who has been trained in definition writing, will produce a solid first draft. This does not mean that the first definition must be the one and only permissible expression of a concept. Since one concept may occur in different contexts and may be described to different user groups, it may occasionally be desirable to have several definitions. Ideally, user-facing definitions meet definition criteria for terminological definitions and can be reused. In any case, this terminology workflow model assumes a conceptual database; consequently, different definitions for the same concept must be part of the same conceptual entry.

Input of secondary sources generally originates from database searches during which the searching party did not find the concept or its designator. The searching party can either be a person, or a tool that extracts terms from a particular document and checks the results against an existing database. While the danger of introducing duplicate entries may be a problem when product unit personnel are entering terms, it is *definitely* a problem when this step is carried out semi-automatically. Few existing tools allow users to avoid duplicate conceptual entries; therefore the experience and skill of the author is key. In most cases, a terminologist is tasked with the research and possibly the authoring of the entry. Nonetheless, the bulk of the input ideally comes from the originators of the concept. This could mean that the terminologist consults directly with the product unit or appropriate subject-matter expert. It could also mean that the definition is based on material located during the research. In essence, the closer the information is to the source, the better.

The value added by this step is obvious: whereas there was no entry before, after this step there exists an entry that contains a wealth of company-specific subject matter knowledge. The knowledge that matters most in this step is subject-matter expertise, but also the understanding of what a term is and when it is stable enough to make documentation worthwhile. Creativity and basic understanding of terminology management are also important factors in this step.

Review and standardization

So far, the knowledge of the individual has been captured but not necessarily formatted for use by others. In practice, most SMEs follow input standards during preparation of their first draft or a terminologist works with the SME to author a good first draft of an entry. This means that the conceptual input and standardization phases overlap in most scenarios. Whether the review phase exists independently or not, an entry must be evaluated in terms of some or all of the following criteria: legal, marketing, globalization, linguistic, content, cross-product, and so forth.

The sequence and scope of these reviews may not always be the same, as they generally depend on the nature of the entry and the availability of resources. Technically, the easiest way to organize such reviews is to attach a workflow routing sheet to an entry and send it off to the individual review instances. The aim is to produce an entry that is sanctioned by all stakeholders with minimum input, but which is reliable and repeatable for the database users. The following paragraphs will describe the nature of the review steps, independent of the order in which they occur.

Experts on marketing, legal, or globalization may spend more time on entries that are related to their area of expertise, and rapidly approve others or not even check them. The marketing department confirms, for instance, that the chosen denominator conforms to the marketing strategy adopted, for instance, vis-à-vis competitors' products. One module of the J.D. Edwards software was initially named "Configurator," but was soon renamed "Visual Configurator," later "Sales Configurator" and "Base Configurator" to call out specific functional aspects. Generally, however, only a subset of terms will require marketing approval. This is also true for legal sign-off. Lawyers and paralegals verify that copyrighted product names, for example, are documented and spelled accurately (e.g., OneWorld™ vs. One World). Copyrighted terms do not need to be checked for localization issues, since they remain untranslated. It is extremely important, however, that they pass a globalization check. During globalization assessment, terms are checked for suitability in other countries. In the German market, the abbreviation "SS" for server system would still evoke the association *Schutzstaffel*, even more than 60 years after the war.

Not all of these checks are performed in every instance and every system, as they depend on the type of term as well as the setting. However, every terminological entry must undergo a linguistic, content and cross-product review. This part of the review has two goals: entry standards must be checked, and furthermore the placement of the concept in the conceptual system of the product, subject-matter area and ultimately the organization must be confirmed.

One important aspect of a functional workflow solution has to do with the status of terms. A conceptual entry might remain "work in progress" for several weeks, if not months. During this time, translators, editors and other researchers may very well use the unfinalized term, but the status of the entry should clearly indicate the relative reliability of the information it contains.

The value added by this step is a proofed entry that meets the standards of all contributing subject-matter experts. In this step, it becomes particularly clear why an automatic workflow pays off. Experts, such as lawyers, will only have to open an entry and, in most cases, give it a quick glance, mark it "approved" and send it on to the next person in the workflow. Although time per entry is low, the value added is extremely high. The knowledge that is most critical here is the appropriate application of written standards, or even laws, in every area of evaluation.

Release

Before the entry is released, one more step must take place, namely a check of the content against the terminology quality standards of the organization as well as against those of the conceptual system. Since previous entry contributors have devoted attention to individual aspects, it is advisable that an authoritative generalist or team of generalists perform global QA on the entry. This QA could entail a spell-check, a check of conformance with regard to definition standards, mandatory fields, cross-entry consistency, possible links, the conceptual system, and so on. At this stage, QA should not require an extensive time investment.

The value added by this step is quality control and verification of conformance to specifications and quality standards. The output of this step is the final entry, which should be logical and not leave any questions unresolved. The person or team performing this review must understand the entire workflow, know enough about the subject matter to discover mistakes, and have an excellent understanding of the entry standards.

Output and use

The output of a terminology management system is a released entry that contains pertinent and up-to-date information, and which can now be used in various output media, such as product-specific glossaries, bilingual term lists, commercially available corporate term banks or dictionaries, online help, etc. In essence, regardless of the nature of the terminology request, the terminology database should be able to handle it.

Although use of the terminology is a largely separate phase, it influences decisions upstream. For example, the choice of data categories might be different if the TMS output is destined for machine translation rather than human translators. Since terminology management has the same philosophical foundation as ontology management, a company may decide to base search mechanisms on the information in their terminology database. In any case, systematic source terminology management is a prerequisite for effective management of target terminology.

Ongoing review and update

As mentioned earlier, even while a term is in process, the entry remains visible to users of the database. The status of the term indicates to the user the relative quality of the term and whether or not feedback may be required before release. After an entry has been released, users anywhere in the organization may leverage the information to create more knowledge. No improvements will be necessary in the majority of cases. However, feedback is of the essence in the case of a small number of entries, as when a subject-matter expert cannot be located, when additional information may exist in the company unbeknownst to the entry's author, or when the conceptual system is not yet clear, for example. Every time a user queries the database and finds an existing entry, certain associations are created. Some are not pertinent and will not result in feedback, whereas others are important and should be captured in the form of feedback to be directed to the terminology team.

Feedback on such small units of knowledge as terminological entries is only provided when a feedback mechanism is readily available. It is extremely important that the feedback tool be user-friendly. For example, the feedback form should be automatically connected to the entry in question. It should also be automatically sent to the appropriate authority, and be clear and efficient.

Once the feedback has been received by the appropriate authority, it must be addressed in a timely fashion. It is best that the requester receive an automated response followed by an update of the status of the request. With every request for revision, the knowledge base grows and the quality of the database is enhanced. It is important to realize that the more languages are handled in a database, the better the source record must be, the more systematic the terminology management approach must be, and the better the designators must be.

The author's experience suggests that if the data is always as up-to-date as possible, if the tool is easy to use, and most importantly, if all terminology is stored in only one place, anyone in search of terminology will use it. Frequent use by people with a variety of backgrounds as well as the possibility of receiving quick feedback will increase the overall quality of the database. If, for example, a term is superseded by another term, but the product team has not updated the entry to reflect that change, the first user of the entry who notices the discrepancy will send a change request to the entry owner, who will promptly add the missing information. Likewise, if a localizer who is translating product information cannot find the equivalent of a concept in his language, he will send a translation request to the

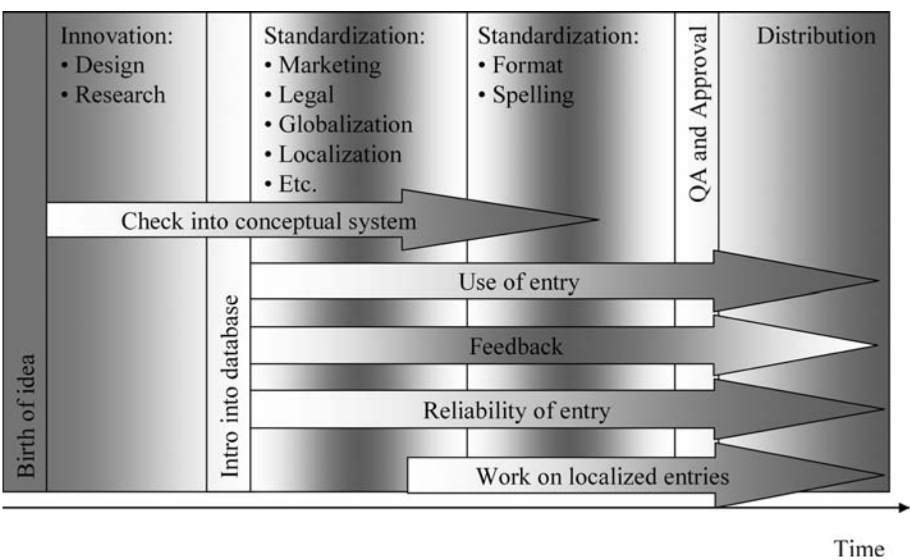

Figure 4. A workflow model.

terminologist and note a suggested translation in the request form. The terminologist will confirm the accuracy of the suggestion and add the entry.

The individual workflow steps are summarized in graphical form in Figure 4. The workflow begins with an idea that is researched, introduced into the database, standardized, approved, and distributed. An important aspect that transcends the individual workflow steps is the integration of a concept into the conceptual system. The darker shades indicate a higher intensity of a certain step or process. Because the entry becomes more reliable as it passes through the workflow steps, it will be queried on and applied more often. Initially, there may be plenty of feedback, but in turn the reliability of the entry increases. Eventually, it is reliable enough to attach a foreign entry to it.

Conclusion

While this discussion focuses on a deployment in a software company, the business process described would not be very different if translation services were subcontracted to an outside vendor. In fact, Bowne Global Solutions, a major provider of localization services, has been using a similar system for a while with its corporate clients (Dettmann 2002). The workflow of the CLS terminology solution at Corporate Language Services also contains comparable steps (see also Fähndrich 2003).

In sum, the most important aspects to consider when evaluating or developing a terminology management system with automatic workflow are the following:

- the knowledge type, including how to explicate implicit subject-matter expertise and the various knowledge types for terminology management;
- the knowledge culture, including the reward system surrounding knowledge sharing and training opportunities;
- the knowledge infrastructure, i.e., the possible forms of technical implementation, supported by standards.

If knowledge management is not so much about creating order or structure as it is about facilitating that ability of organizations to know or to use knowledge (Addleson 2000: 137), then the system discussed here provides a perfect tool. In essence, a terminology management system is a tool for gathering knowledge from all possible places in an organization, standardizing the entries for ease of use, and then reusing and improving the database over time. It is a venue for employees to exchange knowledge and contribute; it is a community of practice (Mazzie 2000: 104) founded upon the improvement of the organization's conceptual and linguistic assets. And since sense-making is a social process, a terminology workflow that enables and facilitates feedback is a perfect tool for this exchange.

Notes

1. The following programs, among others, offer degrees or certificates in terminology management: École de Traduction et d'Interprétation (Geneva), Institut Libre Marie Haps (Brussels), Universitat Pompeu Fabra (Barcelona), Universität Wien (Vienna).

2. For more information on return on investment through knowledge management see Koenig 2004 and Powell 2004.

3. In 2004, the SAP TMS contained 1.6 million main entries in 31 languages, and 23,000 definitions in German and English (Childress 2004b).

4. For Drucker's vision of the new organization see Drucker (1998).

5. Peter Senge popularized the concept of the learning organization in his book *The Fifth Discipline* (Senge 1990).

6. Bowne Global Solutions was acquired by Lionbridge in June 2005.

References

Addleson, M. 2000. "Organizing to know and to learn: Reflections on organization and knowledge management." In *Knowledge Management for Information Professionals*, T.K. Srikantaiah and M.E.D. Koenig (eds), 137–160. Medford, NJ: Information Today, Inc.

Budin, G. 2002. "Wissensmanagement in der Translation." In *Übersetzen und Dolmetschen. Eine Orientierungshilfe*, J. Best and S. Kalina (eds), 74–84. Tübingen/Basel: A. Francke Verlag.

Childress, M. 2004a. "Terminologiemanagement und Wissensmanagement bei der SAP AG." In *Terminologie und Wissensmanagement*, F. Mayer, K.-D. Schmitz and J. Zeumer (eds), 127–143. Cologne: Deutscher Terminologie Tag e.V.

———, M. 2004b. "Terminology at SAP — A brief history." 7th International TAMA Conference: Multilingual Content Integration, Cologne.

Dettmann, D. 2002. "Softwarelokalisierung bei Bowne Global Solutions: Weltweite Terminologieverwaltung in Echtzeit am Beispiel von TermGlobal™." In *eTerminologie — Professionelle Terminologiearbeit im Zeitalter des Internet*, F. Mayer, K.-D. Schmitz and J. Zeumer (eds), 143–150. Cologne: Deutscher Terminologie Tag e.V.

Drucker, P.F. 1998. "The coming of the new organization." In *Harvard Business Review on Knowledge Management*, President and Fellows of Harvard College (eds), 1–19. Boston, MA: Harvard Business School Press.

Fähndrich, U. 2003. "Terminology management solutions: Portraits of problem-solving with internal and external terminology services." Paper presented at the annual conference of the American Translators Association, Phoenix, AZ.

ISO TC 37/SC3. 1999. *ISO 12620:1999. Computer Applications in Terminology — Data Categories. Part 2: Data category registry.* Geneva: ISO.

Karsch, B.I. 2002. "Integrierte Terminologieverwaltung — Die Einführung einer Terminologiedatenbank beim Softwarehersteller J.D. Edwards." In *eTerminologie — Professionelle Terminologiearbeit im Zeitalter des Internet*, F. Mayer, K.-D. Schmitz and J. Zeumer (eds), 165–178. Cologne: Deutscher Terminologie Tag e.V.

———. 2004. "Wissensmanagement und Lernorganisation am Fallbeispiel J.D. Edwards." In *Terminologie und Wissensmanagement*, J. Zeumer (eds), 101–117. Cologne: Deutscher Terminologie Tag e.V.

Koenig, M.E.D. 2004. "Time saved: Not a political justification for knowledge management." In *Knowledge Management Lessons Learned: What Works and What Doesn't*, M.E.D. Koenig and T.K. Srikantaiah (eds), 141–143. Medford, NJ: Information Today, Inc.

Malhotra, Y. 2000. "From information management to knowledge management: Beyond the 'hi-tech hidebound' system." In *Knowledge Management for the Information Professional*, T.K. Srikantaiah and M.E.D. Koenig (eds), 37–61. Medford, NJ: Information Today, Inc.

Martin, B. and Karsch, B.I. 2001. "Terminology management driving content management." Paper presented at the 5th TermNet Symposium: Sharing Terminological Knowledge. Antwerp.

Mazzie, M. 2000. "Key challenges facing the evolution of knowledge management." In *Knowledge Management for the Information Professional*, T.K. Srikantaiah and M.E.D. Koenig (eds), 99–113. Medford, NJ: Information Today, Inc.

Microsoft New Zealand. 2004. "Microsoft New Zealand begins Te Reo Maori version of Microsoft Technologies." Microsoft New Zealand PressCentre. http://www.microsoft.com/nz/presscentre/articles/2004/apr_26_tereomaori.aspx

Nonaka, I. 1998. "The knowledge-creating company." In *Harvard Business Review on Knowledge Management*, President and Fellows of Harvard College (eds), 21–45. Boston, MA: Harvard Business School Press.

Oeser, E. and Picht, H. 1998. "Terminologieforschung in Europa: ein historischer Überblick." In *Fachsprachen — Languages for Special Purposes*, L. Hoffmann, H. Kalverkämper and H.E. Wiegand (eds), 341–347. Berlin/New York: Walter de Gruyter.

Powell, T.W. 2004. "Knowledge return on investment." In *Knowledge Management Lessons Learned: What Works and What Doesn't*, T.K. Srikantaiah and M.E.D. Koenig (eds), 125–139. Medford, NJ: Information Today, Inc.

Schmitz, K.-D. 2000. "Softwarelokalisierung — Eine Übersicht." In *Softwarelokalisierung*, K.-D. Schmitz and K. Wahle (eds), 1–10. Tübingen: Stauffenburg Verlag Brigitte Narr GmbH.

Senge, P. 1990. *The Fifth Discipline: The Art & Practice of The Learning Organization*. New York: Currency Doubleday.

Short, T. and Azzarello, R.C. 2004. "Knowledge management in action: Nine lessons learned." In *Knowledge Management Lessons Learned: What Works and What Doesn't*, M.E.D. Koenig and T.K. Srikantaiah (eds), 31–53. Medford, NJ: Information Today, Inc.

Wright, S.E. 2001. "Terminology and total quality management." In *Handbook of Terminology Management: Application-Oriented Terminology Management*, S.E. Wright and G. Budin (eds), 488–502. Amsterdam/Philadelphia: John Benjamins.

PART 5

Localization education

A discipline coming of age in the digital age

Debbie Folaron

In response to the call for more localization professionals over the past few years from industry, courses and programs have sprung up in different areas of the globe to meet the growing demand for translators, language specialists ("linguists"), programmers, engineers, and project managers able to deal effectively with the specificities of diverse localization environments. As a matter of course, and indeed, much as industry itself has done, academic institutions have proceeded to assess and question the demands and very nature of these relatively new localization-related professions. They have done so in part to be able to justify the creation and funding of projects and courses, and at the same time they have sought to understand if localization is just a passing trend, or a lasting phenomenon. This article addresses some key general issues that emerge from the dialogues on professional localization training and education taking place in academia. They reflect the need to articulate concretely how this new discipline is being shaped and defined, and the necessity to maintain credibility and be accountable to both academic and professional sector demands. These areas will be discussed here in three broad sections:

1. defining the name, terms, and parameters of the discipline;
2. partnering the practical demands of professional industry goals with the intellectual inquiry of academic objectives; and
3. developing curricular modules within the scope of the above.

1. Defining the name, terms, and parameters of the discipline

As is the case with any academic discipline that has not yet enjoyed a long history or been fully legitimized through normalized channels of debate, critique and the production of canonical works, the field of "localization" has experienced initial growth pangs. One of these stems from the fundamental need to more clearly define its name, as well as the principal terms and concepts that constitute it as both a functional field of activity and as an object of analysis. What is "localization?" What is a "localizer?" What and why do we "localize?" Given that many localization courses are attached to or located within translation programs, how does localization differ substantively from translation? Should we consider localization as a specialized sub-set of translation, or is the reverse truer to form and practice? Does the usual simple definition of "linguistic, cultural, technical adaptation of products" suffice? Does this definition necessarily imply parameters and concepts that are solely market-oriented, based only on the languages of business, commerce, and marketing?

The study of any discipline, as a discipline, entails consideration of its practices and theories, its history and current directions, its texts and its acquired body of knowledge. The localization discipline thus far has largely been defined through its practice, one that is complex, multi-disciplinary, and notably beholden to technologies constantly in flux, hence not always easy to pin down. Nonetheless, practice of the profession has unquestionably laid the groundwork and opened the way to defining the parameters and dimensions of this new area of study. In order to clarify the definitions of the name and terms, two dimensions will be considered here: (1) procedural — localization as one component of the larger GILT[1] processes prescribed currently by some major industry players; and (2) historical — localization as it developed within industry, against the backdrop of revolutionary transformations in society and economy. Briefly, a few salient characteristics of the industry itself are first worth mentioning. The world of localization has been competitive and collaborative in nature since its inception. It has been the domain predominantly of digital content creators and publishers, localization service providers, multi-language service vendors, software tools developers and vendors, and multinational corporations. Within the scope of its projects, it has forced traditional channels of translation production to branch out and interact in teams with other professional service groups. It has evolved in tandem with technological development and the implementation of new information and communication technologies across the planet. It operates in a context that is increasingly one of digital globalization, which has affected businesses of all types at many different levels, forcing them to renew and adapt business strategies in order to cope with and succeed in the face of this rapid technological change. It has borne the brunt of a high number of mergers and acquisitions in a trend towards consolidation over the past decade, indicative of a sector that is still in its early growth stage.[2] In essence, the localization industry — tightly embedded within channels and networks of information and communication and modes of production dependent on technology — is dynamic, moving, fast-paced, and in a constant state of flux.

How exactly has "localization" been defined in practice? Consensus as to its basic definition has often proven elusive. Descriptions and perspectives on what constitutes localization, its content and procedures, have emerged sporadically from diverse professional domains and have tended to vary depending on: (1) one's position in the localization chain (translator or programmer, for example); (2) training (language, translation, computational linguistics or computer science); (3) experience (projects for large multinationals or for smaller multilingual service providers); and (4) geographical region. For the first programmers and engineers who were working on the U.S. west coast writing code and developing English-language software that would ultimately require adaptation for international markets, localization inevitably referred to the introduction of a time-consuming and "foreignizing" linguistic-cultural phase into their usual routine chain of domestic technical programming, writing and production. Moreover, this phase was sometimes an unwelcome addition mandated by marketing and management. Applications and corollary support information painstakingly had to be transferred — technically, linguistically, and culturally — from a U.S./American English environment into a multiple-platform, multilingual, multicultural environment. For translators and translation agencies, on the other hand, it meant the introduction and implementation of computer-assisted translation (CAT) tools and technologies, combined with certain software engineering abilities, into the usual workflow of linguistic-cultural transfer, merely to be able to manage the new content and its format effectively. It entailed mastering both the message and the (new emerging forms of) medium. To complicate matters, the definition of localization has tended to vary considerably in accordance with the needs of projects and the client localization infrastructures in place.

A good bare-bones working definition, and one that allows us to further qualify and expand when put in historical context, has been advanced by the recognized localization authority Bert Esselink (2003: 4): "localization revolves around combining language and technology to produce a product that can cross cultural and language barriers. No more, no less." Esselink rightly notes that it is the marriage of language and technology that created localization, and which is continuing to define the changing scope of localization, although, as we shall see, it seems that this definition could apply to related globalization and internationalization activities as well. The variable term in this definition is "technology." The rapid proliferation and ever-increasing complexity of new computer, information, communication, and Web technologies continue to provoke profound transformations in companies and organizations that generate source content for subsequent professional translation. They require businesses to constantly innovate technologically, and to rethink their strategies and adapt their workflows in order to remain competitive. By the same token, the new technologies have impacted conventional processes of linguistic-cultural transfer in translation, and created a new technological dialectic for translators. They have come to play a major role in reshaping translator tools and practices on one hand, while causing an explosion of translation volume in a dizzying array of formats on the other. The transforming variable that is technology has progressively caused the processes and products of translation and localization to be redefined, and in some cases even to overlap. This fact alone problematizes any facile or stable definition of the terms of the field.

That said, one of the most recent terms, the acronym GILT (Globalization, Internationalization, Localization, Translation), seems to synthesize localization and interrelated and interdependent activities of the profession most succinctly. While it does not, in actual practice, represent the phases all companies or organizations undertake when embarking on a localization project, it does reflect the main phases manifest in the universe of processes that comprise the current localization landscape. Unpacking the acronym in order to understand it, however, entails knowing a few historical facts. Translation (T), of course, is generally understood to be the basic linguistic-cultural transfer of source text to target text. It has existed for as long as humans have chosen to write and needed to communicate across linguistic and cultural borders. Localization (L), as mentioned previously, was a term coined in the late 1980s by software developers to reflect the introduction of linguistic-cultural elements considered foreign to the initial source code, content and display in U.S./American English. It has essentially referred to the linguistic and cultural adaptation of content for different locales according to local, regional, and national customs, standards and conventions, enabled through technology. As time has passed, attempts have been made periodically to distill the essence of the term. Currently, high-profile industry associations such as the Localization Industry Standards Association (LISA) and publications such as *MultiLingual Computing & Technology,*[3] respectively propose a general definition of localization as follows:

> Localization: [T]aking a product and making it linguistically and culturally appropriate to the target locale (country/region and language) where it will be used and sold (LISA 2005).
>
> The process of adapting a product or software to a specific international language or culture so that it seems natural to that particular region. True localization considers language, culture, customs and the characteristics of the target locale. It frequently involves changes to the software's writing system and may change keyboard use, fonts, date, time and monetary formats (MultiLingual Computing & Technology 2004b: 5).

Localization was confined primarily to the area of software applications and accompanying technical documentation for the first ten to fifteen years of its history, accelerating in volume and intensity with the mass commodification of personal desktop computers in the mid to late 1990s. In its second phase, it moved into the domain of Web sites and Web-based applications, a tangible result of the commercial explosion of the Internet, World Wide Web, online services, and high-speed communications infrastructure. Publication and regular updates of digital content on the Web in multiple languages has subsequently brought localization into the realm of complex multilingual project management. The Excellence in European eContent Localisation project now defines localization as "adapting to processes that support the creation and management of multilingual web content, together with the web architectures required for multicultural environments" (EEEL Online), and Reinhard Schäler articulates it as the "full provision of services and technologies for the management of multilingualism across the digital information flow" (2003: 102). As observed by Folaron and Vesler (2003: 37–40), the term can be broadened further to include complex multilingual projects for local ethnic and minority communities residing within one nation-state, official language and culture, and government. Finally, it is worth noting that localization has expanded out of the domain of proprietary, commercial projects to embrace projects that are open source, as manifest in discussions on the Digital Divide Network (digitaldividenetwork.org), the International Open Source Network (iosn.net), and in forums and conferences focused on open source localization, such as the Open Source Localisation Conference, organized by the Localisation Research Centre in September 2004. Esselink (2003: 4–7) underscores this changing nature of localization projects as follows:

> Even though typical software localization projects may still be the bulk of the work for many localization service providers, they are quickly being supplanted by new types of localization projects … Also, content translation projects are now often considered as localization projects simply because of the complex environments in which the content is authored, managed, stored and published.

The characteristically changing nature of localization projects leads us back to our acronym. The complex preparation necessary for multilingual localization has proven, by experience, to be more cost-effective and time-efficient for some companies and organizations when combined with a localization-enabling phase known as Internationalization (I), itself part of a larger more encompassing process known as Globalization (G). Hence, the objective has become one of implementing a more pro-active strategy from the start, rather than defensively combating expensive complications as they arise, particularly when localization is dealt with as an afterthought to the process. In other words, the strategy to globalize is part of preparation for localization. In theory, a company or organization first commits to carrying out its activities on a global scale or in the global arena, and then (assuming that it has a clearly articulated global sales, marketing and business strategy, which unfortunately is not always the case), devises business, management, or organizational strategies at the outset that will most effectively deal with issues that inevitably arise outside the local perimeter where it is based. A salient example that comes to mind is compliance with legal regulations for foreign financial, medical, or accounting applications. The strategy to internationalize — as part of preparation for localization — is complementary, and it refers to the process of designing the outgoing source content to be as linguistically, culturally and technically neutral as possible so as to facilitate subsequent localization(s). Linguistically, this may mean adhering to controlled language style guides and terminology while writing the source content. Culturally, this may mean avoiding graphics or images (with or without embedded text) in the source content that are heavily dependent on local cultural referents. Technically, this may mean creating source code that supports international natural language character sets, contains added functionalities specific to foreign locales, and separates translatable strings from the actual code base. Let us defer once again to LISA and *MultiLingual Computing and Technology*, which propose the following general definitions for globalization and internationalization:

> Globalization addresses all of the enterprise issues associated with making a company truly global. For the globalization of products and services this involves integrating all of the internal and external business functions with marketing, sales, and customer support in the world market (LISA 2005).
>
> [Globalization a]ddresses business issues associated with launching a product globally, such as integrating localization throughout a company after proper internationalization and product design (MultiLingual Computing & Technology 2004b: 5).
>
> Internationalization is the process of generalizing a product so that it can handle multiple languages and cultural conventions without the need for redesign. [It] takes place at the level of program design and document development (LISA 2005).
>
> [Internationalization is t]he process of generalizing a product so that it can handle multiple languages and cultural conventions without the need for redesign (MultiLingual Computing & Technology 2004b: 5).

As new contexts and work environments materialize, we can expect GILT definitions to specialize and diversify even more. New terminology emerges as a result. The exponential growth of game localization is one recent phenomenon, bringing into the mix highly complex technical capabilities, gaming dynamics, and some skills once considered to be the exclusive domain of creative literary and audio-visual translation. In the domain of wireless technologies, for example, the term "homologation," was introduced to the readership of *Localisation Focus* in the March 2004 issue as follows:

> Homologation: the testing and certification of a product or specification to indicate that it meets regulatory standards. Services include explanation and interpretation of standards and specifications, assistance in audit and approval, testing and certification, product design consulting, translation of manuals, legal mandates and other material (Deignan 2004: 6–8).

In sum, the field of localization needs to be defined in terms of its own historical dynamics and its growth as an industry constantly absorbing new technologies, as well as through its relations to the peripheral and interrelated phases of globalization, internationalization, and translation. In this way, a clearer picture of its specificities comes into focus. We see that all GILT processes — translation, localization, internationalization, and globalization (GILT) — address, albeit in different ways, diverse language and culture issues. We see that conventional translation processes transferring linguistic and cultural content must accommodate technological transfer too, and that translation emerges as just one operational link in the chain of target end content production. When set against the backdrop of a networked, global information and knowledge economy, this technology-enabled transfer of linguistic-cultural content increasingly stands out as one of socio-technological adaptation, with referents that hail to the means and processes of information and knowledge organization within given societies, to issues of usability and functionality, and to social interaction.

Technologies will continue to directly influence and nurture the professional practice of localization. The academic field of localization can complement and supplement this practice by serving as a window through which to explore different subjects of inquiry, raising issues and questions beyond the scope of localization training proper. For example, given that practical localization activities depend so heavily on the economy and on technologies, one might interrogate and deconstruct the very foundation of the global information economy that feeds these activities. One might research, in the context of global macroeconomics, how the technologies implicated in localization have served to both hegemonize and fragment, uniformize and diversify, distinct sectors and interests over the globe. One might also consider, as does Manuel Castells (2000; 2004), how technology and technical relationships of production have organized, permeated and diffused throughout the whole range of social relationships and social structures. How do the contradictory forces to globalize/uniformize and localize/diversify manifest themselves in the so-called network society, or knowledge-based economy? Is there a new technology paradigm? These questions clearly transcend the actual field and discipline of localization practice. Nevertheless, as we see in recent scholarship, they serve as points of departure or as contextual referents for contemplating localization as a larger cultural practice. Some have already begun to do so. For translation scholar Michael Cronin:

> the very term 'localization' begs the question as to whether we have a translation practice that is unifying — disseminating software originated in US English throughout the globe — or fragmentary, that is, highlighting local identities and differences. The question then is whether localization is to be classified as translation-as-homogenization or translation-as-diversification (2003: 86).

For Reinhard Schäler, the question is:

> how to "preserve [linguistic and cultural] diversity while removing the barrier for a more equal and inclusive information society … [and] while creating products that use globally acceptable content," since "languages and cultures will always give us access to different worldviews — a fundamental reason for preserving their integrity" (2003: 102).

While these kinds of questions and propositions currently cannot be defended or sustained when making the business case for concrete decisions on budgets and available resources of localization projects, they do serve to encourage awareness and alternatives to directions already in place. They might inspire sorely needed research on the specificities of foreign or international markets and on users of localized products. They also reflect the need to have the humanities engage in meaningful dialogue with business and technology, which might ultimately translate into significant or meaningful input in the processes of localization. As localizable content becomes more multilingual, multicultural and digital in nature, and is generated using an ever-expanding range of tools, technologies, and media, academics will be challenged and inspired by a wide range of subject areas for research, from concrete translation-for-localization practice (such as the translator use of parallel — i.e., multilingual — corpora, for example) to more socio-cultural oriented commentary on translation and localization process (such as, for example, the consequences of outsourcing within the context of globalization).

In conclusion to this first section, then, we see important questions emerge as a consequence of the need to define the field. How to develop sustainable academic courses and programs (whose proponents often invest and expend considerable time and energy as their proposals and plans make their way through the requisite bureaucracy and channels of approval) when the fundamental definitions of the profession appear to shift, even if slightly, depending on where, when, and by whom they are defined? How to ensure that academics can manage to keep abreast of the rapidly changing technologies and tools, and of professional workplace practices? Should academic departments be responding solely and defensively to immediate market demands, focusing on training students and upcoming professionals on the tools and technologies alone as their only desirable goal? By the same token, is it not a defensible argument to insist that graduates be exposed to the technologies and business dynamics and realities driving localization? "There is still," as noted by the EEEL project, "no widespread, common understanding of the issues and challenges of localization" (EEEL Online). Indeed, if academia seeks to educate, enlighten, influence or empower those who will be the principal participants in society and in this professional sphere in years to come, there is a great deal that can (and should) be done. In addition to providing the necessary training and education needed to succeed in the localization industry, we can strive to promote reflection on the discipline on another level. We can thoughtfully analyze the interactive interfacing of humans with machines, and society with technology; we can mediate trends to automate and processes of creativity so they learn to cohabitate more productively; and we can keep alive the vision of cultivating greater representation and participation of all linguistic, ethnic, cultural, geographic, and technical groups in the global environment.

2. Partnering the practical demands of professional industry goals with the intellectual inquiry of academic objectives

Defining the name, terms, and parameters of the localization profession and discipline is a vital first step. Establishing a general profile that sketches and fulfills both professional and academic objectives is equally important. In fact, until now the backgrounds of localization professionals have run the gamut from comparative literature to software engineering to computational linguistics. Drawn into the field for a variety of reasons, they trained and learned on the job during the early formative years of the industry. They have since contributed to writing about the profession and its needs[4] as the industry endeavors to reflect upon itself and on the procedures it employs. The skills demanded in professional job listings and projects still consistently attest to the fact that localization projects continue to vary widely in scope and breadth, ranging from the localization of stand alone desktop applications and accompanying desktop published user manuals to highly interactive dynamic database-driven Web sites and accompanying multimedia tutorials, and on to games and mobile phones. Which of the skill sets required for this environment, then, could possibly intersect with the skills and knowledge that more traditional or established academic disciplines have sought to foster? Can the stringent fast-paced demands of professional industry goals effectively cohabit with the measured pace of intellectual query and method synonymous with academic rigor? In the interest of creating and sustaining fruitful and mutually beneficial dialogue between the professional world of practice and academic discourse, can we find any common meeting ground? First, I would argue that many of the cognitive skills we desire to encourage in traditional humanities and liberal arts courses (including translation programs) do find their counterparts in localization. Second, localization, like translation studies, transcends disciplinary boundaries and is thus conducive to insightful interdisciplinary academic investigation.

A. Cognitive skills

With the aim of introducing and cultivating this common meeting ground between industry and the academy, I would first argue that certain higher-order cognitive processes and aptitudes are appreciated equally by the localization industry and by institutions of higher education. These skills include: conceptualization; abstraction; research; analysis; synthesis; comparison; strategic application; critique; and formulation of original thought. These skills refine and sharpen our abilities to think, reason, query, establish cognitive relationships, conceptualize and articulate ideas, organize content and procedures, and inspire creativity. As college and university mission statements and syllabi routinely testify, they are the desired outcome of a general liberal arts education, hence valorized as academic discipline skills. As industry job descriptions likewise regularly testify,[5] they are desired qualifications for work in localization or GILT-related domains, hence valorized as professional skills.

There is already a great deal of existing literature (academic and professional) on the nature and definition of knowledge, knowledge representation, cognition, and cognitive aspects and approaches with regard to "process" on its diverse levels. Within the context of general arts and humanities programs, we foster acquisition of knowledge through the application and honing of cognitive skills. This encourages deeper analysis of process, ranging on the continuum from creation to critique. For example, in the traditional domains of literature, music, art, and translation as well, we speak on one hand of the composition and de-composition of a whole work into its constituent parts through an assortment of techniques, devices, perspectives, approaches, and methods, and on the other hand of the relationship of this work to broader historical, social, political, and cultural contexts. Cultivating cognitive skills of this type allows us to substantially increase our appreciation of how all the singular aspects interact with one another, and how they produce the whole we contemplate at hand: in terms of its functionality, its ability to engage with our senses, its aesthetic appeal, its usefulness to us, and its meaning to us. Subsequent conjugation with historical, social, political and cultural contextual factors will unmask and generate a vast and multifarious range of relationships. These kinds of approaches and methods all underscore the critical importance of grasping the relationship between part and whole, of understanding process, and of establishing meaningful relationships within broader contexts.

I believe that professional localization projects and practice, through their unique combination of interdisciplinary components linking and interfacing technology, language, culture, business and management in a globalizing environment, harbor a potentially rich and fruitful terrain (not yet fully documented) on many different levels for exploring diverse cognitive aspects of interest to academic research. At the same time, these cognitive aspects and relationships — in particular the processing of information — are of concern to professional practice itself, for they give substance and form to the practice of localization by virtue of the specificities of the respective disciplines interacting with one another during the process. A few examples will serve to illustrate this point. Computer science and artificial intelligence — embracing such domains as logic and reasoning, the representation of knowledge, and natural language processing — envisage the machine (computational system) as an information processing system conceptualized and designed to represent certain types of knowledge. The concepts underlying programming, relational databases, and neural networks in these disciplines, for example, have impacted the content, tools and technologies implicated in diverse phases of the localization process. Human–machine interface research has expanded to include not only the domain of software but also the Web, where the parallel evolution of knowledge representation and information retrieval systems is yielding such structures as the Semantic Web. Communication studies — which analyzes, among other things, the components that constitute "message" and the (multi-)media by which this message is transferred and interpreted- envisages production and reception as an information processing system, intended to communicate certain types of knowledge. Concepts configuring the occasionally overlapping textual, visual, tactile, and audio grids of this system have likewise impacted the content, tools and technologies implicated in diverse phases of the localization process. Translation studies, benefiting from psychology and linguistics, envisages the human mind as an information processing system wired for learning certain types of knowledge, for synthesizing structured and experiential knowledge, and for interpreting and transferring meaning from one language to another. Concepts underlying human translation, and machine translation as well, have impacted the content, tools and technologies implicated in diverse phases of the localization process, in particular as human information processing interfaces more and more with machine information processing.

All of the above consider the part/whole relationship in terms of the process(es) and broad contexts that motivate and activate cognition. Demonstrable mastery of the part/whole relationship becomes tangibly and critically important for localization and complex multilingual projects specifically with regard to management. Project integrity must be maintained throughout the entire process. This means that all related issues, problems and objectives must be considered conceptually both as a whole and in the details. This mastery and management of the relationship between part and whole entails visualizing, understanding, and maintaining vigilance over the big picture, while managing linear and overlapping phases, workflow procedures, and the interaction of various team tasks, functions, and roles. Much more than mere mechanical processes and file-moving, or impressive juggling acts of mindless multitasking, genuine management in localization projects is the application of high-level cognitive skills to practice, without which source content transfer cannot be effectively accomplished. These same skills apply at the stage of source content creation, be it for software or the Web. As applications, systems, information architectures, and approaches to design become ever more varied and complex, higher-order cognitive skills are increasingly crucial for maintaining conceptual and functional integrity (from idea to representation to design to deployment) of all the interfacing facets and phases. These skills must be fine-tuned to the degree that one can concentrate wholly and separately on issues and problems at both the levels of concept and detail, i.e., on how individual components synergistically comprise the "whole," and how the overall "whole" is designed to function and perform by means of its composite parts. In line with basic principles of cognitive instruction, localization courses afford students ample opportunities to connect thinking skills and problem solutions. Students can learn to select and implement strategies, based on the knowledge of how, when, and why to use them in given situations and projects, and to ultimately develop creative solutions themselves. Cultivating these cognitive skills to better comprehend how the complex whole interacts with its individual parts allows teachers to impart a greater sense and appreciation of "process." A curriculum that focuses on process and which fosters self-sufficiency as well as collaboration will ultimately empower students by allowing them to integrate more readily into the professional world, without having to undergo extended training periods in order to be productive in a commercial environment.

B. Interdisciplinarity

Common ground between professional practice and academic discourse can be expanded through the interdisciplinarity that characterizes localization. Beyond the practical training on specialized tools and technologies or technical savoir-faire normally associated with it, localization practice reflects a unique convergence of disciplines: foreign languages; linguistics; translation; computer science, desktop publishing, graphic design and layout; and international business; to name but a few. The concrete practical experience of authentic projects and case studies grounded within the multilingual, multicultural, multimedia, multi-technological spheres that increasingly typify localization projects cannot help but contribute favorably to the kind of interdisciplinary cross-fertilization that has always caused academic disciplines to thrive. Some general themes and subjects particularly conducive to academic investigation in the humanities and which are peripherally, if not directly, connected to the diverse processes at work in the world of localization include:

1. society in terms of its dynamics, infrastructure, and historical narrative;
2. histories of technologies and their impact on societies, including that of information architecture (Harris and McCormack 2000: 33);
3. conceptualization and representation of knowledge;
4. dynamics of human communications, specifically in terms of a digital environment (including virtual team collaboration; virtual communities; open source projects);
5. dynamics of human organizational structure;
6. dynamics of human commerce (including diverse cultural and commercial conventions; commoditization of information in relation to knowledge, communication and content management technologies and services; customized mass consumption, etc.);
7. dynamics and dimensions of human play, and its diverse cultural manifestations (including varying degrees of instructional and entertainment motives, as witnessed in the global game industry);
8. faces of globalization (economic, technical, cultural, linguistic), and their manifestation in ideologies and power relations; the "digital divide";
9. transforming notions of nationalism;
10. cultural expression and self-representation in diverse societies;
11. construction of "identity";
12. representation of the "other";
13. history of world languages and the study of languages in contact (including their status);
14. position of translated/localized content (or localization itself) within the cultural-socio-economic dynamics of a continually globalizing environment;
15. specific linguistic, historical and political experiences of a region or country in terms of how they determine language and culture-related policies and political decisions, and as a consequence, translation and localization processes (in fact, do these processes ultimately enrich or alienate, emancipate or dispossess, the languages and cultures being represented?) (St-Pierre 1998: 47–56);
16. relationships between languages, both natural and machine; natural language processing and machine logic (including machine translation — MT — processes of analysis, transfer, and synthesis); and digital representation of natural language writing systems (phonetic and ideographic systems);

17. study of human–machine interface, in terms of the personal computer, Web, voice technologies, and mobile phone technologies (including short message service — SMS — and multimedia message service — MMS);
18. comparative text and discourse analysis, for printed materials and for the Web;
19. multimedia (graphics, audio, video, animation, text) analysis;
20. writing and linguistic/cultural adaptation for diverse genres of communication, including controlled language;
21. business and management concepts and tasks, including process planning, resource management, progress tracking, quality control, compliance with industry standards, team management, assessment and consulting (MultiLingual Computing & Technology 2004a);
22. ethics.

In well-established disciplines, theory and practice create a working relationship that seeks to mutually benefit professional work and academic discourse. A detailed overview and analysis of how this might materialize for localization is beyond the scope of this article, but some points for thought are worth raising. Any given discipline rightly seeks out the salient and unique aspects that manifest its differences in relation to other established disciplines, and then endeavors to reflect on its own nature and processes with the double goal of articulating itself more comprehensively and eliciting new perspectives on itself as a subject of inquiry. Any field that is highly interdisciplinary in nature regularly discovers approaches and methodologies from related or interrelated fields which when applied to itself serve to elucidate and transform existing ideas. Translation studies is one such field. Translation theory in particular has enjoyed and continues to enjoy a symbiotic relationship with a number of diverse disciplines, including linguistics, history, social sciences, philosophy, comparative literature, cultural studies, foreign languages and literatures, and computer science. It has profited from the heightened awareness of the nature and ramifications of contact between languages and cultures issuing from a multilingual globalizing environment. On-the-ground, real-life cases of translation in practice have provided substance to theory in the form of linguistic and cultural details. Likewise, translation theories have meticulously examined source and target texts, in terms of what constitutes a text and of how translators proceed to process text and embark on translational transfer with the goal of achieving "equivalence" and "communication." They have discussed the influence and effects of translated works (including selection; decision to translate; target audiences; position of translation within structures of production, etc.) in society. They have interrogated the notion of ideology in translation and examined the dynamics of power relations through translation theories informed by literary, critical and cultural theory. They have taken a critical look at such fundamental concepts of human existence as identity, representation, and culture.

As the engaging ways theory and practice have informed one another in translation theory and in other interdisciplinary academic disciplines, so should the interaction between theory and practice prove to be fertile terrain on many levels for localization studies. Through socio-cultural and politico-economic approaches, for example, theories of localization could explore professional commercial localization motivated by profit and other business imperatives. They could research localization projects born of geopolitical realignments and restructuring, such as by the recent integration of central and eastern European countries into the European Union (Safar 2003). They could broach questions of identity, power relations, and representation within the multilingual, multicultural dynamics of Web site localization. They could even investigate the effects on content of localization project managers selecting in-country professional translators who have been uniquely impacted by national language policies, inter-language tensions on national territory, and colonization. Localization is very much about understanding and preparing content for the context and environment in which the translated/localized product will be received. It deals with linguistic registers and discourse; multicultural symbolic and iconic representation; diverse writing systems; world and local legal systems, legislation, and jurisdiction; cultural and commercial conventions; intellectual property, and so on. Translation theories have shown us that the receiving target context is *a priori* grounded in linguistic, cultural, historical, and political specificities and experiences that will influence translators and translation principles, strategies, processes and practice in general. Localization theories might make fruitful contributions to these studies and to the multilingual, multicultural and multimedia perspectives already contemplated by translation theories by examining and problematizing the multi-technical, multi-technological layers and dynamic team interface of localization practices.

Localization theories could also pick up the relay from translation theories that study text and translational transfer to consider the features, qualities and general constitution of source content that is processed through localization. How does content directed through the whole GILT process compare with content localized as an afterthought? They could examine the form and content (or "packaging" and content) of "message" — which is at the heart of the act of communication, and the object of translational transfer — as it is defined by the medium it constitutes. They might study localizable and localized content as part of the digital environment, in which "content" is defined comprehensively as "any digitized information — that is, text, document, image, video, structured record, script, application code, or metadata — used to convey meaning or exchange value in business interactions or transactions" (DePalma 2003: 6). They might examine, contingent on the definition of "content" presented in *The Sapient Report*, specific aspects of the ways in which the communicated message interacts and engages with the receiver(s), i.e., as "a system of words, images, audio and video that is integrated with information architecture and visual design to communicate the brand" (Harris and McCormack 2000: 9).

The content may be static, but it may just as well be dynamic,[6] and as such, refer and defer signifiers differently for each interpreter-participant on the receiving end. How does this fact affect the goal of replicating the source experience in the target-language content? Likewise, how can a document written for the Web be considered in terms of its structure, i.e., as a set of entities, with start and end tags that delimit the content or which can contain attribute values? Does structuring content so it can be processed computationally or viewed in Web browsers fundamentally alter the way an author conceives and applies ideas to form? Is creativity in fact stimulated and generated in the process of creation simply by virtue of being able to visualize ideas and experience forms in WYSIWYG applications, in which "What You See Is What You Get?" Is the process of creation more open-ended and non-committal because an author may easily, and rather autonomously, modify and transform at will the content that was initially put to form? Is dynamically generated content or a document for the Web always in the state of becoming, rather than finalized, and what does this do to our notion of aesthetics? How is this problematized by localization practices? How do the selection of languages and the entire GILT process affect the targeted audience? Are there varying degrees of translational and localizational acceptance for diverse cultures or for the same group of language speakers within a designated region? How do members of target culture read localized content? Are Internet users (monolingual; bilingual; multilingual) who browse the Web reading localized content any differently than readers of translated content in traditional print? How do responses to these questions affect multilingual project management, and translation strategies, at the levels of globalization and internationalization (if at all)?

Additionally, localization theories might also expand on studies of ideology and power relations down through the production chain. They might descriptively explain the position of localization and localized content (both for software applications and multimedia Web-based content) in society. They can problematize these positions by asking how access to the Internet or to code that is free and open source qualifies the terms of accessibility and acceptability by the public. What delimits "elite" from "popular," and acceptance from marginalization, when we speak of the digital world? Can we even apply a dialectical model of cultural critique or has the global landscape been transformed in such a way that the terminology previously used to critique cultural phenomena no longer suffices? Are there useful criteria we can extrapolate from previous historical models of cultural or social critique when we consider digital modes of production, digital transfer, and source and target content that can be accessed only through digital means? What effect does the intervention of localization practices have on all this? What are the tangible effects of diverse international, national, regional, and local policies, histories, and trends on the sectors being targeted for localization, from both source and target perspectives? SAP explains on its Web site, for example, how local legal and financial policies on the Latin American continent directly affected linguistic, cultural, and technical transfer (localization) processes at the source point of origin (SAP INFO 1999). The Indian Ministry of Communications & Information Technology, to cite another example, noted at a recent localization conference how India's multilingual (18 official languages), multi-script (10 different scripts) situation requires that software and content not only be localized from foreign languages to Indian languages, but between Indian languages as well (Vikas and Chopra 2003: 17–19).

Finally, while acknowledging the significance of theories that ensue from the interaction of existing disciplines with one another (particularly in light of their socio-cultural, ideological, political, and economic dynamics), it is equally important to engage localization as much as possible on its own terms. Kersten, Kersten and Rakowski (2001: 11), for example, discuss the different degrees of culture manifest across different application domains in terms of "surface" and "deep" cultural components. They argue that applications based on real-world social interaction (an e-business framework, a Web-based customer relationship management and a Web-based banking interface) can only be intuitively understandable if the applications are patterned on a cultural model that mimics local cultural interaction. Core functionality, then, must be based on culture-bound relationships and assumptions within a given society. For this, they propose "software culturalization," which would "extend … the concept of software internationalization to the business logic of applications" (Kersten, Kersten and Rakowski 2001: 11). We might usefully extrapolate this relationship between cultural practice and functionality to the domain of video, computer and Internet game localization for the global market. Like so many other disciplines, localization needs theory and practice to create a mutually beneficial dialogue and working relationship between professional work and academic discourse. This implies an intellectual and professional ethical responsibility to understand the diverse realities of localization in practice.

3. Developing curricular modules within the scope of professional industry goals and academic disciplinary objectives

Establishing a general profile that takes into account both professional and academic objectives is important when considering the ways in which academic institutions can ally industry goals with disciplinary aims in a localization program of study. Currently, specific "localization" courses and programs vary widely throughout the world.[7] They are often set up in accordance with local and international market needs. Many are separate certificate programs housed within Continuing Education divisions; most are attached to translation programs. Others are specialized courses offered in seminars and workshops. Most focus on practical training and have established productive links with industry and the private sector in order to remain current in the field.

In terms of creating and designing curricula for full-fledged academic programs, localization might draw some inspiration from the extensive work already carried out in the field of translation studies over the past fifty years (Schäffner and Adab 2000).[8] One angle that seems particularly fitting and adaptable to localization as a discipline is that of "process," which assumes that the final product "whole" is the result of a series of complex interacting individual processes which merit being broken down, studied and mastered separately such that a coherent and cohesive whole can once again be constructed, from source to target. Since the professional localization environment is as much about process and modularity as it is about the end product, a productive educational environment can be created in which elements of practical training tasks and intellectual academic exercises are developed synchronously and creatively through a process-rich context, organized loosely in terms of "competences" as desirable outcomes.

Although the notion of "competence" and how it can be achieved and evaluated is still under vigorous debate in both the professional world and academia, advocating a process-oriented competence approach does, I believe, provide a general base on which to begin discussing key terms and structuring a flexible framework for localization curricula design within an academic infrastructure. It underscores performance, problem identification and problem solving, and is consequently dependent on choices, strategies, and solutions for transfer of content and the production of a high-quality end result. This is in line with industry expectations. It likewise finds precedents in translator training literature. Most translation scholars would agree that competence generally lies between the recognition and acquisition of basic features and concepts of the linguistic/cultural transfer that constitutes translation (including, for example, knowledge of the source language-culture and target language-culture, subject area, text and genre types, and transfer strategies), as well as the proficiency and expertise needed to apply creative strategies to choices in order to solve a vast array of translational problems for a variety of text types and genres aimed at diverse local publics. On the learning curve, it can be considered the phase at which aspiring translators comprehend the standard processes and goals of translation, and where they are applying world and subject knowledge, socio-cultural context, resources, technologies, models, acquired skills, and common sense to devise practical and creative strategies for identified problems, all with the goal of producing an acceptable translation (Adab 2000; Chesterman 2000; Neubert 2000).

Cognizant of these precedents, and of the fact that translation competence tends to focus on individual translators and translational transfer from source to target text rather than on team- and project-based transfer from source to target content (as is the case for localization), we can tentatively propose the following competences for categorization. Based on multilingual localization project management and diverse team specializations, they constitute three basic categories: (1) Management; (2) Technology; and (3) Language-Culture.

Localization competences

Competence 1: Management

Understanding of:

- standard GILT processes;
- the components (phases, tasks, teams) of a localization project, in terms of various models (in-house, outsourced, or various combinations thereof) and of desired client outcomes;
- how to assess the degree of "localizability," i.e., the extent to which the source text is written so as to be adaptable for local markets, and in an internationalized file format that enables easy extraction and reinsertion of translated text (Ingram 2004: 7);
- how to assess the degree of localization necessary, i.e., full or partial, based on concrete end-user needs;
- how to evaluate "content" and "product" (functional user application vs. pure information);
- how to analyze, evaluate, classify, prioritize, and manage various levels and types of information;
- how to identify tasks and problems (including potential problems) methodically, and how to propose choices and solutions based on education, knowledge and experience;

- how to apply world and subject knowledge, socio-cultural context, resources, technologies, models, acquired skills, and common sense to processes;
- how to apply standard and creative strategies to tasks and problems with the goal of achieving superior communication transfer;
- how to create project plans that take into account tasks and potential problems (i.e., risk management), with options that are standard and creative;
- how to manage synergy between diverse teams (for team deliverables) including program or content engineering and design, desktop publishing and/or Web design, translation and editing, terminology;
- management of translation memory (TM) and/or any other CAT tool (including MT);[9]
- how to communicate and negotiate effectively;
- how to create and implement quality control procedures;
- how to assess the project in terms of quality control, compliance to standards, and testing;
- how to make the business case for localization and frame the salient issues in actionable, business-critical terms.

Competence 2: Technology

Understanding of:

- the basic concepts that concern data: its creation, structure and organization, packaging, management and retrieval;
- what constitutes a database (fields, tables, records, etc.) and how data is applied to multilingual applications (Jewtushenko 2003: 19);
- what constitutes (in terms of current technologies) a "document,"[10] or, as discussed earlier, "content";
- the technologies used to create source content and the techniques for separating localization-related source content data from non-localization-related data (extract-localize-merge paradigm), since non-separation is often an issue in cases of incomplete globalization or internationalization;
- basic relationships: parts to whole; data objects; elements and attributes; variables;
- information structures, including the packaging, unpackaging, and repackaging of information;
- diverse types of workflow;
- current tools and technologies, so as to be able to analyze the formats of project source content and to assess which tools, technologies and expertise will be needed to manipulate that content throughout the project;

- available resources and back-up solutions so as to be able to apply (or to have applied) creative strategies in the absence of expected functionality in terms of CAT and localization tools, as well as the applications and content being localized);
- the dynamics that underpin basic information, content, communication, and Web technologies, including human–computer interaction and ergonomics;
- the technologies implemented to translate and localize content generated by and for diverse systems, including document management systems, mobile devices, automated workflow and enterprise applications (content management systems, globalization management systems, knowledge management systems, enterprise resource planning systems, etc.), Web-enabled applications of all types, database systems;
- the creation and manipulation of content based on standards[11] supported by the World Wide Web Consortium; XML; XLIFF; TMX; TBX; OLIF; OSCAR; and Open Tag in general (see Appendix);
- character sets and encoding, in particular Unicode;
- internationalization and internationalized source content in technical terms of character encoding, locales, local conventions, cultural issues and their application early in the design phase, prior to localization;
- translation, linguistic/cultural adaptation of content in accordance with a system's locale settings, including decimal number, time, currency, and measurement formats, as well as sorting, spelling, and hyphenation, so the end user can process information in his/her language and script without loss or corruption of data (Schäler 2003: 106–109);
- the processing of digital content formatted for diverse languages and scripts, and encoded for diverse platforms and peripherals;
- the following areas of specialization:
  - programming and engineering for software applications, in particular to facilitate subsequent localization;
  - writing for the Web (document structure, and meta and markup languages; SGML, XML, HTML, and others; text formatting; fonts; display);
  - creation of multimedia components and applications;
  - database creation and management (including terminology and translation memory databases);
  - desktop publishing;
  - CAT/MT/L10N tools (in particular the principles on which the tools and technology functionalities are based);
- technologies as they are applied to translation.

Competence 3: Language-Culture

Understanding of:

- localization history in terms of the development of computer software programming, Web coding and mark-up for different linguistic families in diverse geographical areas;
- languages in terms of culture and contact with diverse linguistic-ethnic groups;
- economic globalization brought about by evolving information and communication technologies, and interacting markets, governments and financial institutions;
- network-based organizational structures and cultures that constitute the information economy and/or knowledge economy on a global scale;
- controlled language and source content authoring for subsequent translation/localization;
- localization and internationalization consulting;
- the implementation of at least two levels of cultural adaptation: "general" (colors, images, signs, symbols, sounds, history, terms and acronyms, etc.) and "radical" (entertainment, education and training, information and its dissemination, content display and retrieval, cultural attitudes towards individualism, collectivism, institutions of power, authority, gender, tolerable behavior, problem-solving approaches [top-down vs. bottom-up, for example], acceptability standards in advertising, and humor).[12]

While daily professional work in this fast-paced field does not generally give one the time to retreat and contemplate all the ongoing processes at hand, a more thorough sensitization to and comprehension of workplace dynamics and interacting processes through education can surely enhance the professional experience and stimulate insight and creativity. Curricular objectives based on the general competence areas suggested above can facilitate integration of the aims of both theory and practice. Informed and enriched by an understanding of the history, skills and goals that comprise both professional and academic visions, they seek in a more cogent way to attain excellence in the linguistic, cultural and technological transfer of source to target content that is at the heart of localization itself, and the core and essence of what it represents: communication and the exchange of fundamental information in the global digital age. With globalization having profound repercussions on so many aspects of people's lives across the globe, from trade and commerce to immigration and human rights, and with the global translation and localization market projected to reach $11.5 billion by 2007,[13] the educational stakes are clearly high enough to warrant immediate attention. Localization shows every sign of being far more than a mere passing trend.

Notes

1. G (globalization); I (internationalization); L (localization); and T (translation).

2. As this article goes to press, two major industry acquisitions have occurred: SDL International acquired TRADOS, and Lionbridge acquired Bowne Global Solutions.

3. Organizations include LISA (www.lisa.org), GALA (the Globalization and Localization Association: http://www.gala-global.org/en/index.php), TILP (The Institute of Localization Professionals: http://www.tilponline.org/). The Localization Institute (http://localizationinstitute.com/) and the Localisation Research Centre (http://lrc.csis.ul.ie/) coordinate seminars, workshops, and summer institutes. The Localization World conference is held bi-annually with venues alternating between Western Europe and the United States (http://www.localizationworld.com/). *MultiLingual Computing & Technology* publishes industry news and information on its Web site (www.multilingual.com).

4. One tangible attempt to compile resources and information in an ongoing, up-to-date manner is the *Annual Localisation Reader*, fruit of collaboration between *MultiLingual Computing & Technology* and the Localisation Research Centre in Limerick, Ireland: http://lrc.csis.ul.ie/LttNWeb/index.htm; http://lrc.csis.ul.ie/LttNWeb/annual_localisation_reader.htm (The Localisation Teaching, Training and Research Network). Bert Esselink's *A Practical Guide to Localization* has become a *de facto* textbook (see http://www.locguide.com/). Likewise, the EU-funded eCoLoRe project, with resources for curricula, was launched in early summer 2004 (http://ecolore.leeds.ac.uk/xml/project/news.xml?lang=en).

5. Austin Community College (Texas) has developed a Localization Generalist certificate program, and provides links through their site to job offers and descriptions (http://www.austincc.edu/techcert/more_localization_info).

6. "Dynamic" here refers to "(1) Web pages containing dynamic content (e.g., images, text, form fields, etc.) that can change/move without the Web page being reloaded or (2) Web pages that are produced on-the-fly by server-side programs, frequently based on parameters in the URL or from an HTML form" (http://en.wikipedia.org/wiki/Dynamic_Web_page).

7. Localization-related courses are filed in ELECT online (http://www.electonline.org/learning.php).

8. This includes the (sometimes tenuous) relationship of professional requirements to academic infrastructure, and how this materializes in translation programs at different institutions around the world (Anderman and Rogers 2000).

9. Jeff Allen regularly writes about CAT, L10N, and MT tools and technologies and their implementation in diverse businesses and organizations (http://www.geocities.com/jeffallenpubs/localization.htm).

10. As noted earlier, much has been written on the subject of "text," and it would be interesting to explore more deeply the notion and changing dynamics of "document."

11. As noted, standards can be definitions or formats approved by a recognized standards organization or accepted as a *de facto* standard by the industry. Standards affect various aspects of design, development, and testing.

12. Terms used by Reinhard Schäler and John Malone (Archetypon), with "general localization" focusing on superficial cultural differences like language, currency formats, date and time; and "radical localization" focusing on cultural differences that affect the way users think, feel, and act, including learning styles" (Schäler 2003: 87).

13. Michael Klinger, Globalization Division Manager at Venturi's Globalization Division cited the following statistics from Allied Business Intelligence during his presentation at The National Language Conference (2004):

- $11.5 billion global human translation market by 2007
- $134 million of this market will be in machine translation
- $3.4 billion in software localization
- Web site localization, the fastest growing segment, from $499 million in 2001 to $3.1 billion by 2007 (CAAG).

Game industry research predicts that worldwide online game revenues will reach almost $10 billion by 2009 (DFC Intelligence 2004), another rapidly emerging market for localization.

References

Adab, B. 2000. "Evaluating translation competence." In Schäffner and Adab (eds), *Developing Translation Competence,* 215–228.

Anderman, G. and Rogers, M. 2000. "Translator training between academia and profession: A European perspective." In Schäffner and Adab (eds), *Developing Translation Competence*, 63–73.

Bowker, L., Cronin, M., Kenny, D. and J. Pearson. (eds). 1998. *Unity in Diversity? Current Trends in Translation Studies*. Manchester: St. Jerome.

Castells, M. 2000. *The Rise of the Network Society*. Vol. 1 of *The Information Age: Economy, Society and Culture*. Oxford: Blackwell.

———. 2004. *The Power of Identity*. Vol. 2 of *The Information Age: Economy, Society and Culture*. Oxford: Blackwell.

Chesterman, A. 2000. "Teaching strategies for emancipatory translation." In Schäffner and Adab (eds), *Developing Translation Competence,* 77–90.

Cronin, M. 2003. *Translation and Globalization*. London/New York: Routledge.

Deignan, J. 2004. "What on earth is H10N?" *Localisation Focus* (March): 6–8.

DFC Intelligence. 2004. "The online game market heats up." June 30. http://www.dfcint.com/game_article/june04article.html

DePalma, D. 2003. "Rage against the content management machine." LRC '03: The 8th Annual Localisation Conference and Industry Showcase. Limerick, Ireland: Localisation Resource Centre. http://www.localisation.ie/publications/presentations/2003/Conference/Presentations/DePalma%20LRC.ppt

Esselink, B. 2003. "The evolution of localization." *The Guide to Localization*. Supplement to *MultiLingual Computing & Technology* 14 (5): 4–7. http://www.multilingual.com/FMPro?-DB=vendors&-lay=CGI&-format=gettingStarted/welcomeGettingStarted.htm&-find

EEEL Online (Excellence in European eContent Localisation). 2002 [2002–2005]. "Localisation industry at the crossroads." http://www.eeel-online.com/app/eeeldes.nsf/eeelookup/ionLocalisation+industry+at+the+crossroads?Opendocument&m=true&t=Aboutloc&

Folaron, D. and Vesler, I. 2003. "Managing translation for communities in New York City." *MultiLingual Computing & Technology* 14 (4): 37–40.

Harris, J. and McCormack, R. 2000. *Translation is not enough. Considerations for global Internet Development*. Cambridge, MA: Sapient Corporation. http://www.sapient.com/pdfs/strategic_viewpoints/globalization_a4.pdf

Ingram, C. 2004. "Working with the client to ensure a successful project." *The Guide to Project Management*. Supplement to *MultiLingual Computing & Technology* 15 (3): 7–9. http://www.multilingual.com/FMPro?-DB=vendors&-lay=CGI&-format=gettingStarted/welcomeGettingStarted.htm&-find

Jewtushenko, T. 2003. "An introduction to XLIFF." http://www.localisation.ie/publications/presentations/2003/Conference/Presentations/LRC_XLIFF_overview.ppt.

———. 2004. "An Introduction to XLIFF." *Fourth International LRC Localisation Summer School*, 59–91. Limerick, Ireland: Localisation Research Centre.

Kersten, G., Kersten, M.A. and Rakowski, W.M. 2001. "Software and culture: Beyond the internationalization of the interface." *Journal of Global Information Management* 10 (4): 86–101. http://kerstens.org/mik/publications/softwareAndCulture-jgim2002.pdf.

Klinger, M. 2004. "Venturi's globalization division." *Proceedings of The National Language Conference.* http://www.nlconference.org/docs/9A%20NLC_mikeklinger.ppt

LISA. 2005. "Frequently asked questions." Localization Industry Standards Association. http://www.lisa.org/info/faqs.html

MultiLingual Computing & Technology. 2004. *The Guide to Project Management*. Supplement to *MultiLingual Computing & Technology* 15 (3).

———. 2004. "Basic terminology." *MultiLingual Computing & Technology* 15 (6): 5.

Neubert, A. 2000. "Competence in language, in languages, and in translation." In Schäffner and Adab (eds), *Developing Translation Competence,* 3–18.

Safar, L. 2003. "Localization in central and eastern Europe." *MultiLingual Computing & Technology* 14 (7): 15–18.

SAP INFO. 1999. "Localisation policy continued." *SAP INFO* (November 11). http://www.sap.info/index.php4?ACTION=noframe&url=http://www.sap.info/public/en/article.php4/comvArticle-193333c63b49187313/en

Schäffner, C. and Adab, B. (eds). 2000. *Developing Translation Competence*. Amsterdam/Philadelphia: John Benjamins.

Schäler, R. 2003. "Cultural adaptation of products." *Third International LRC Localisation Summer School*, 102–119. Limerick, Ireland: Localisation Research Centre.

St-Pierre, P. 1998. "Theory and practice: Translation in India." In Bowker, Cronin, Kenny and Pearson (eds), *Unity in Diversity? Current Trends in Translation Studies*, 47–56.

Vikas, O. and Chopra, P. 2003. "Localisation in the context of bridging the digital divide: An Indian initiative." *8th Annual Localisation Conference and Industry Showcase*, 17–19. Limerick, Ireland: Localisation Research Centre.

PART 6

Localization standards

Localization standards, knowledge- and information-centric business models, and the commoditization of linguistic information

Arle Lommel

Localization, the process of adapting products and associated materials for specific markets, is often thought of as a recent endeavor, an outgrowth of the development of the consumer software industry beginning in the 1980s, and is usually considered distinct from translation *per se* in three respects: first, it has a commercial emphasis on *products* and associated collateral rather than simple *texts*; second, localization often focuses on the non-linguistic cultural aspects of a product; third, localization often involves the use of computer-assisted translation (CAT) tools. It should be noted that none of these conditions are necessary conditions, and that even when all three criteria are met, the boundary between translation and localization is often quite fuzzy. While certain translation tasks, such as translation of literature or interpretation, are not usually considered localization, precisely defining what constitutes localization can be somewhat difficult.

For a particularly clear illustration of the differences between localization and translation, consider an automobile designed in Japan but sold in the U.S. market. Prior to release in the U.S., the automobile and its collateral materials will be radically altered in many ways: printed text materials will be translated (or even completely rewritten) for a primarily English-speaking audience, and possibly for a Spanish-speaking audience as well. Informatics systems will be re-engineered to reflect the cultural norms of the U.S., possibly using different types of voice for speech components, or changing how information and alerts are presented to the driver. The automobile will also be altered in substantial physical ways, with the steering column moved from the right to the left side to suit U.S. roadways. In some respects, automobiles represent the extreme end of localization: most products do not require major physical re-engineering, but because they represent the extreme, automobiles are a good example of how localization differs from translation. Few people would claim that the U.S. model of the car was a "translation" of the Japanese version, at least in a common-sense understanding of the term. What

Figure 1. 1895 advertisement for industrial lathe parts announcing the availability of English, French, Spanish and German catalogues.

is clear, however, is that all the adaptations revolve around the preparation of a specific product (an automobile) and associated collateral for a specific *locale* (a linguistic, political and cultural area).

While the rise of the software industry has no doubt been a leading factor in the development of a localization industry that serves to prepare products (usually software and associated collateral such as manuals) for specific markets, translation has long played a key role in the facilitation of international commerce. An advertisement from an 1895 issue of *Scientific American* advertises the availability of marketing collateral (catalogs) in four different languages, and is very similar to any modern advertisement that mentions the availability of multilingual materials (see Figure 1).

While no one would have called these catalogues in French, Spanish or German "localizations" — they would have been "translations" at the time — this advertisement shows that at least one aspect of modern localization — translation of product collateral to facilitate the sales of products in "foreign" markets — was occurring over a century ago. Even the languages and markets chosen by the Westcott Chuck Company (French, German and Spanish) are still the typical first target languages for localization from English. (Unfortunately, the author has been unable to locate copies of these catalogs and cannot comment on the amount of cultural adaptation apparent in them for overseas markets, so it is impossible to say whether the second defining characteristic of localization mentioned above was met in this example.)

If localization (or something very much like it) has been going on longer than is commonly assumed, why then has the issue of standards in localization or translation become a major issue only in the past few decades, and reached high visibility only in the last ten years? What has caused standards to become a topic of considerable business importance and interest? Two major trends are driving the increasing focus on standards: the first is the increasing abstraction of information from presentation made possible by computers and electronic storage of data (in this sense the simple model of computer/software-driven localization is actually fairly accurate), particularly with the development of Extensible Markup Language (XML), which facilitates information exchange; the second is the emergence of *information* as a business commodity with tangible and real value — a proposition that is not as self-evident as it might seem.

These two trends are strongly linked: without the ability to abstract *knowledge* and reuse it as *information* that can be readily manipulated and repackaged, knowledge cannot be readily commoditized; that is, it cannot be treated as a fungible and reusable resource with concrete value. Although information has always had value for businesses, the tendency to view it as having value in itself is not universal nor of particularly ancient vintage.

While steps towards the business commoditization of textual and other information began with the earliest forms of writing (which is often thought to have arisen from recording of commercial transactions), only the digitization of text in abstract forms (such as "plain text" and, more recently, XML) allows reuse of textual information in arbitrary forms, something that Content Management Systems (CMSes) try to capitalize on. The value of information depends in part on the cost to create it, and also on the value it acquires when it can be reused in novel contexts, thus eliminating the cost to recreate information for each context and purpose.

As used in this article, *knowledge* is considered to reside in individual humans, while *information* is the external medium for the exchange of knowledge. (This usage differs from many other formulations, such as Akoff 1989, and the terms used here apply only to this article even if the concepts may be more generally applicable.) When information can be abstracted from the level of human-to-human interaction or from concrete instantiations, issues of standardization become important. Information abstraction and information commoditization are the framework in which this article will explore efforts at standardization carried out by the Localization Industry Standards Association (LISA), and to a lesser extent the Organization for the Advancement of Structured Information Standards.

The Westcott Chuck Company advertisement (see Figure 1) provides a starting point from which to view these trends. In this advertisement, information is abstracted to a certain extent: it has been put in print, and thus does not depend on a single individual for dissemination. Nevertheless, the information is still not terribly abstracted: it is tied to a physical medium (a plate set by a printer somewhere), and cannot be reused in another format without physically recreating the information. Thus the catalog in German could not be easily transformed into a sales leaflet or a brochure, for example. Because it cannot be readily abstracted, the value of this information in any given form is relatively low: it cannot be reused or repurposed, and has the primarily *instrumental* value of selling a product (a drill chuck in this case). The value within this system of knowledge and information dissemination lies not within the information itself, but within its ability to accomplish something else.

In the case of a modern company that makes use of computers and associated information technology products, the equation is radically changed: while the modern company might place an advertisement in the back of *Scientific American* that would be similar in many regards to the advertisement from over a century ago, the advertisement would represent one instantiation of abstracted and reusable information. The same information used to generate the advertisement might be used to produce a Web page and might be copied and pasted into a brochure. If the company uses a content management system (CMS), the text might be abstracted to the sentence or paragraph level and reused in on-line knowledge bases, accessed by support staff, and reused for other related products. If the information is translated and the translated versions linked in some manner to other translated versions, this translation and linkage represents an even greater level of information abstraction: the information is no longer tied to a particular language, but really resides in an interlingual space since there is no longer a single formulation of the information, but rather multiple formulations that have different degrees of authority, the source-language version being considered the most authoritative, but not the only, representation of this information.

This shift toward the increasing abstraction of information can be represented schematically as shown in Figure 2.

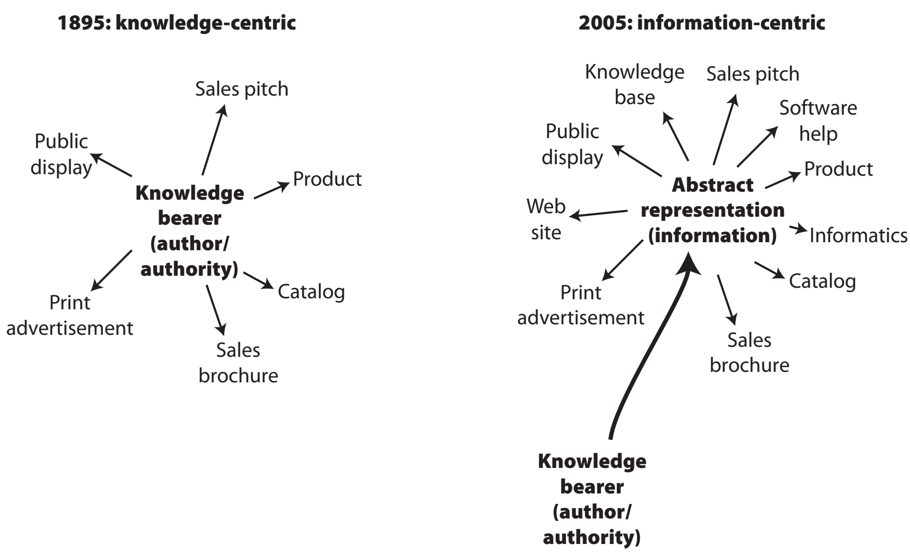

Figure 2. Shift from a knowledge-centered to an information-centered business model.

In the earlier model, information is at the periphery of the model, and knowledge is at the center: a knowledgeable individual produces the various forms of information, whether by writing something or by speaking or engaging in public display of a technology. In the later model, the abstract representation of knowledge as format-independent information has assumed the central role, and the knowledge bearer is on the periphery: once information has been abstracted, the knowledge-bearer may have no further role. He or she may leave the company or move on to other projects, with little or no impact on the information itself. Moving to an abstract information-centric model also impacts the form of some of the instantiations of information: while in 1895 a public display would have required a knowledge-bearer's active presence, in 2005 it is not uncommon for an individual with little substantive knowledge of a product or service to use a PowerPoint presentation (produced from an information archive of abstracted knowledge) to make a public display on a product. In addition, information in any one of the information instantiations can, in principle, be easily converted to any other form (e.g., through copying and pasting text on a computer), something that would have been impossible in 1895. The information has thus acquired a potential value in re-use that depends on its abstraction from any one particular use.

Information abstraction, change management and localization tools

It is in the context of the radical shift in the production and dissemination of knowledge from a knowledge-based system to an information-based system that the *modern* localization industry was born, and the localization industry is a fundamental part of this shift: by taking knowledge recorded in one language and making it available in other languages, information is further abstracted and issues of representing the information become more and more critical.

It may sound paradoxical to discuss the creation of multiple language-specific instantiations of information as abstraction since each translation/localization is in fact a rendering of information into a format (a language in this case) that is not that abstract. However, the move towards localization is a move away from the specific realm of *a language* to the general realm of *languages*. Since information must always exist in a signifying system of some sort, information will always be in a specific format. However, through localization, information as an abstract proposition is given more specific instantiations and thus wider applicability. While any given end user will need information in a specific language, it is possible to conceive of information as existing outside of any specific language by being available in many languages, as is the case in a multilingual "knowledge management" system.

Another important motivator for standardization in localization is that information is changing, with revisions, new versions and other changes brought about for a variety of factors. The need to maintain consistency between linguistic representations of information is one factor that led to the rise of technologies such as terminology management and translation memory (TM). Because information is commoditized and has value in and of itself (and has a cost associated with it because it is abstracted and maintained), businesses have a vested interest in maintaining consistency, in reusing information and in reducing the costs of maintaining that information.

In modern localization operations, TM is the primary enabler for cost reductions by maintaining multiple representations of information. TM tools form a database that aligns source texts and target texts in units of corresponding informational value. For example, if an English text says *This box contains one screwdriver* and its Hungarian equivalent says *Ez a doboz egy csavarhúzót tart*, then the information is abstracted from one language to two languages through the linkage of a TM database.

A typical TM database would contain thousands of such linkages of information representations. If the information is subsequently revised to state *This box contains one large and one small screwdriver* (even if nothing else in the text from which the database was created is changed), the TM tool allows the Hungarian to be updated to *Ez a doboz egy nagy és egy kis csavarhúzót tart*, maintaining the language-independent abstraction of the information by making sure that the informational value of both languages is maintained and (theoretically) equivalent. When texts are revised, TM thus allows companies to know what portions of the information repository must be updated, and which can be ignored because they have not changed. Without the abstract linkage of the TM database, this reuse of previously translated information would be essentially impossible.

The problem with this view of translation as abstraction is that even as the TM database contributes to the further abstraction of knowledge by removing it from linkage to one specific language, it simultaneously works against abstraction because the information must be encoded in a manner suitable for a specific TM tool. The information is no longer in its most abstract state, but is instead instantiated in the format of a specific translation tool and is thus no longer directly usable by other software products without an import and export routine. Crucially, this limitation also precludes use of the data generated by a particular translation tool within other translation tools, and fosters dependence on a particular TM tool for access to the information within the database.

Thus, one consequence of the shift towards an information-centric model of business is that dependence upon a particular translation tool places the core of a company's information assets (at least in the multilingual/multinational arena) in a position of vulnerability. In a very real sense the company's information assets are held hostage to the activities of the translation tool: if the tool developer goes out of business, the company can be left with a system that cannot be updated to support evolving requirements; if the tool developer decides to develop the tool in a way contrary to the interests of the company, the company may be left with little help to meet its own needs. Other factors also put companies at risk: if two companies that merge happen to use different TM tools to represent their legacy information, merging their information into a coherent whole may well be a daunting challenge.

The business value of translation memory and standards

So just how big is the business issue for companies involved in localization? According to research conducted by LISA in 2004 (Lommel 2004), companies involved in localization use TM tools for an average of 7.2 target languages, and the databases of a typical TM user contains approximately 350,000 source segments (typically sentences). This would mean that, subtracting the source language, a typical TM repository might represent a total of just under 2.2 million target-language segments. The number of words per segment was not addressed in the LISA study, but a rough figure of ten words per segment will be assumed for purposes of estimation, yielding a figure of 22 million words in a TM repository. Assuming for purposes of comparison a per-word translation cost to the company of USD 0.20/word, a typical company involved in localization might have more than USD 4 million of information tied up in a given translation tool. Given that not all text is translated into all languages, a more realistic figure might be around USD 2 million, still a substantial investment in the production and maintenance of information.

At an extreme end, some companies have as much as 100 million source segments in TM databases, and localize into thirty or more languages. Using the same method of calculation as before, such a large-scale user of TM tools could conceivably have TM assets *worth more than USD 3 billion (3,000,000,000)*. Not all TM data is created equally, however, and some segments should be considered "garbage" — outdated or inaccurate segments that nonetheless remain in the database. Information can be rendered obsolete for any number of reasons, and one major task for managers of translation memory databases is pruning outdated data from the database to increase accuracy and improve performance. In the multilingual realm, a frequent source of data corruption is the failure to update databases to reflect final revisions of translated text. In the absence of appropriately changed segments, the database will contain outdated versions of translated text that may actually have an associated cost for companies since they will need to be corrected repeatedly, in each unique target-language instantiation. When the value of these garbage segments is subtracted, the overall value of the information will decrease somewhat.

The actual numbers themselves are really beside the point, since the real value of multilingual corporate information cannot be objectively and quantitatively calculated on a universally accepted basis. In addition, these figures consider only the cost to produce and maintain the information, not its potential positive business value in sales, service and support. The point of these figures is to show that the value of the information contained in a TM database exceeds the cost of the tools themselves by several orders of magnitude, a fact that makes independence from a specific tool highly desirable.

In the current context, in which access to valuable information is controlled by comparatively low-valued tools, two options exist to make sure that data is universally accessible: the first is for everyone to use the same tool or format, and in so doing, to establish a *de facto* standard; the second is to develop a format-neutral standard.

To some extent the localization industry has adopted the first model. As of 2004, 71% of companies that use TM tools make use of TRADOS® products — although not necessarily exclusively (Lommel 2004: 12) — and the developers of some other tools have developed ways to convert TM databases from their internal format to TRADOS' format and vice versa. TRADOS' products thus enjoy a status similar to that of Microsoft Word® in the word-processing market in that they serve as the most widely accepted form of TM, and other products adapt to TRADOS. (In July 2005, SDL, a major competitor of TRADOS in the TM tools market, acquired TRADOS. SDL had long been one of the most public supporters of the TMX standard and had a TMX-based strategy to lure TRADOS users to switch to SDL's tool set. The impact that this acquisition will have on the TM market and TRADOS' position remains to be seen since SDL has not yet publicly revealed its intentions for the two tool sets it now owns.)

The drawback to the *de facto* standard is that it is in the best interest of the company controlling the *de facto* standard to foster dependence on the product that serves as the standard and raise barriers to prevent migration to other competing products. *De facto* standards thus tend to be unstable over time. If the *de facto* standard tool implements new features or changes functionality, workflows that depend on the ability to work with the *de facto* standard are easily broken. The producer of the tool that has become the *de facto* standard has no business obligation to help its competitors (quite the contrary), and meaningful long-term standardization is at best difficult in this model.

Creation of an independent vendor-neutral standard is the logical alternative. In this model, a standard format is developed independent of any single company (although individual companies may be heavily involved and have a stake in the outcome since they must implement the standard). This is the approach taken by the OSCAR (Open Standards for Container/Content Allowing Re-use) group within LISA, and by the XLIFF (XML Localization Interchange File Format) and Trans-WS (Translation Web Services) groups within OASIS.

OSCAR standards

Founded in 1997, the OSCAR group is LISA's standards special interest group. When it was founded, OSCAR's initial mandate was to provide two standards, one for the tool-independent representation of TM data and the second for the tool-independent representation of structured terminological data. OSCAR began working on the first task immediately, and decided to postpone the second until the first task was well under way.

TMX — Translation Memory eXchange

The result of the first task was Translation Memory eXchange (TMX), now considered the grandfather of localization-specific standards. Work on TMX under the direction of Franz Rau of Microsoft and Alan K. Melby of the Brigham Young University Translation Research Group (TRG) was very rapid, and included participation from the following companies: ACM, AlpNet, IBM, ILE, Indigo, ITP, Logos, Microsoft, Multiling, STAR, Systran, and TRADOS (Rau 1997). (Several of these companies have since been acquired or have merged with competitors and thus no longer exist.) Version 1 of TMX was ratified by the LISA General Assembly in 1998.

The OSCAR Steering Committee decided early on that any standards it developed should be XML-based. At the time work commenced on TMX, XML was not yet formally released as a standard, so development of TMX was truly cutting edge. TMX is now considered one of the first true XML-based standards to have been formally released by any standards body. Since its initial release, TMX has been updated to reflect new issues and needs, and is now at version 1.4b. Because of the inherent conservatism in the standards process, TMX 1.4b differs only slightly from TMX 1.0, and the standard has been very stable over time. At this point TMX is also a certifiable standard, with an established process for compliance verification. As of May 2005, the products of three companies (SDL, GlobalSight and Idiom Technologies) have been certified as compliant with TMX 1.4b, and a number of other companies have announced their intention to seek certification. (A current listing of certified products is maintained at http://www.lisa.org/tmx.)

There has been some disagreement within the localization community about the precise role of TMX. Some companies and individuals have worked with TMX as an exchange format (i.e., a format used to migrate data from one format to another) while others have used it as an interchange format (i.e., a format used to provide interoperability between tools). The precise intended use of TMX in this regard was left rather vague by OSCAR from an early date — partially because it was unclear what could be accomplished with TMX — although there was widespread agreement that, at the very least, it would serve as an exchange format for data migration, even if it were not used to provide interoperability.

Because different tools handle markup of source documents differently, version 1.0 of TMX provided a format-neutral way to preserve format markup within TM segments, but it did not provide a specific standard way to represent that markup. In essence, it allowed developers to "escape" markup, and left it up to the receiving tool to interpret that markup. TM tools can thus ignore markup in TMX segments, or attempt to interpret it. At present, no tool offers a comprehensive solution to the problem of markup transfer, and most tools tend to import and treat TMX segments as plain text, a factor that limits the simple reuse of TM data to a certain extent. (Even in cases where markup interpretation is possible, some users choose to ignore markup in order to better facilitate TM lookup and matching based on purely linguistic features.)

As TMX began to be implemented, it was quickly discovered that an issue not initially addressed by OSCAR could (and did) have a large impact on the ability of translation tools to use TMX generated by other tools. Not all tools segment text in the same way. Let us consider the following text as an example:

Pressing the red button empties the tank; pressing the green button loads the tank.

This sentence-level segment contains a total of 83 characters (including spaces). Whereas Tool A might treat this text as one segment, bounded by a full stop/period, Tool B might divide it into two segments at the semicolon. Thus, a TMX file created by Tool A would contain one 83-character segment, whereas the corresponding TMX file generated using Tool B would comprise two segments, each containing 41 characters. If Tool B imported a plain TMX file generated using Tool A, it would not find a match for the above segment during translation, since the source segment in the TM database (i.e., the entire sentence) would be twice as long as either one of the two colon-delimited portions of the source segment presented for translation.

According to trials run by J.D. Edwards (since acquired by PeopleSoft) and reported on to the OSCAR steering committee, differences in segmentation can result in the loss of several percentage points of effectiveness in TM leverage. In other words, up to several percent of 100% match segments may not be found when using an imported TMX database generated by one tool with another tool. The precise amounts vary depending upon the tools used to create and import the TMX information, but 100% matches are the most valuable matches since, if a TM database is well maintained, they require no manual translation and minimal quality control compared to newly-translated materials. In the case of a typical TM user with about USD 2 million invested in a TM database, these few percentage points' worth of lost 100% matches could translate into a direct cost in the tens of thousands of dollars to retranslate material that the TM tool should have been able to retrieve and reuse. In the case of the largest users, the direct costs of retranslation could rise into the millions of dollars. Such sobering statistics give new meaning indeed to the phrase, "lost in translation."

The issue of segmentation has not yet been entirely resolved, and most users have had to accept that TMX does not enable lossless transfer of data between tools, but that there is a data transfer penalty in the process. However, OSCAR turned to the creation of a new standard to address this precise issue.

SRX — Segmentation Rules eXchange

Because of the previously discussed loss of TM effectiveness when TM data is ported between tools using TMX due primarily to differences in segmentation methods between tools, OSCAR formed a working group to address the issue of segmentation standardization. The solution was the development of the SRX (Segmentation Rules eXchange) format, a standard way of representing the rules used to generate TM data. Using SRX, a tool that does not use semicolons as a segment boundary could specify that it does not use semicolons. With an SRX file from the first tool, the second tool could know that if it splits on semicolons, it will miss some 100% matches, and adjust its behavior accordingly. In many cases, making some relatively simple adjustments to segmentation settings within the receiving tool to match those used in the tool that generated the TMX file could eliminate many (but perhaps not all) of the losses experienced in using TMX to port data between tools.

SRX was formally approved by OSCAR as a standard in 2003. At present few tools have integrated SRX, but interest in SRX is high. In 2004 LISA research, SRX was considered the most important development in TM technology by those surveyed, and was the second most important development to the largest users (Lommel 2004: 18–19). In the future, as more tools adopt TMX and are certified as TMX compliant, it is anticipated that SRX will be widely adopted as the best way to deal with cross-tool segmentation issues.

TBX — TermBase eXchange

The second of OSCAR's original mandates was to develop a standard for the exchange of structured terminological data. TermBase eXchange (TBX) was not initially developed by OSCAR because it was recognized that TBX would rely on work being conducted by ISO (International Organization for Standardization) Technical Committee 37/Subcommittee 3 on an XML standard based on MARTIF (MAchine-Readable Terminology Interchange Format — ISO 12200) and the terminological data categories of ISO 12620. OSCAR decided to adopt the results of the SALT project (SALT 2003) as the basis for TBX, and SALT officially turned its results over to LISA for continuation as the basis for TBX. OSCAR and LISA officially accepted this donation and released the results as TBX 1.0 in May of 2002.

In contrast to TMX, industry adoption of TBX has been slow. Three factors seem to contribute to the low implementation of TBX:

1. Compared to TMX, TBX is a complex format, and implementation is correspondingly more difficult; termbases are very heterogeneous in their design and implementation, and using TBX to effectively port terminological data requires an understanding of the structures of both the source and the destination tool.
2. Few companies have any awareness of the value of controlling terminology and so have invested little in structured terminological resources. Without terminological resources to start with, few companies are interested in exchange.
3. Finally, there has been no good way to mark up source documents with pointers to terminological resources, limiting the ability to tie terms within documents to appropriate terminological resources.

LISA and its standards special interest group OSCAR have responded to these three issues directly. The issue of complexity is being addressed through TBX Lite, a simplified subset of TBX that will serve the needs of most casual users (currently being developed by the LISA Terminology Special Interest Group, or TermSIG). The issue of (lack of) corporate awareness of the value of terminology is also being addressed by the TermSIG through the development of educational materials and reports that demonstrate the business value of controlling terminology. While this is not a standardization issue *per se*, the TermSIG efforts have produced increased interest in terminology issues among LISA members. Recent work from LISA has demonstrated that problems with terminology can actually have a major negative impact on the brand and product loyalty of customers (Lommel 2005: 1, 8, 51–52), and this finding may help to establish the business value of terminology standards. Finally, the issue of linking source documents to terminological resources is currently being addressed by OSCAR in the TBX Link specification (released for public comment in May 2005), under the direction of Alan K. Melby (Brigham Young University) and Andrzej Zydroń (xml-Intl Ltd.). Using the XML namespace capability, TBX Link will provide unambiguous pointers to TBX terminology resources and allow them to be leveraged during the localization process.

Although TBX faces a steep challenge in the implementation arena, it is in some ways the most mature of the LISA standards, since it is based on work that goes back decades within ISO. With proper industry education and the development of TBX Lite and TBX Link, TBX is posed to become a major standard. One area of anticipated application in the future is within the Semantic Web (Berners-Lee, Hendler and Lassila 2001), which at present exists primarily as a monolingual conception, but which will need to be extended to a multilingual world with the development of suitable structured terminology.

GMX — GILT Metrics eXchange

The final current LISA standards initiative to be discussed in this context is GILT Metrics eXchange (GMX), released as a draft OSCAR standard for public comment in May 2005. GMX arose in response to the fact that different tools calculate volumetric information about texts in different manners. (GMX is actually divided into three portions, GMX-V for volumetrics, GMX-C for metrics about the complexity of a localization task, and GMX-Q for metrics about the quality requirements of a localization task. This discussion focuses only on GMX-V because, as of the time of writing, work had not yet begun on GMX-C or GMX-Q.)

Different methods of calculation can have a significant impact on the economics of localization since most localization tasks are calculated on the basis of per-word cost. Experiments conducted by the author in 2001 revealed that major localization tools differed by as much as 30% in their word counts, depending on the nature of the source text. (The largest differences were found in tabular texts and depended on whether numbers were counted as words or not, but even in running prose, the differences could be as high as 10–15%.) Some tools took a very conservative approach to word count, while others gave very high counts. Microsoft Word is often used as the basis for word counts in the localization industry, but it counts essentially any content bounded by white space as a word, regardless of the nature of the content. When different tools count words differently, transparency of per-word costs is inhibited. For example, a company that charges USD 0.15/word using a tool that delivers a count of 10,000 words for a given text would actually be more expensive than a company that charges USD 0.17 per word but calculates a total word count of 8,600 words for the same text (USD 1,500 vs. USD 1,462), even though the per-word rate would suggest that the first company is less expensive.

GMX-V specifies a number of verifiable text-related counts that are far more detailed than the traditional word count metric. Using the XLIFF format, it defines a normalized form for texts to be counted, and includes separate counts for numbers and textual elements, as well as a character-count alternative for Chinese, Japanese and Korean, where character counts are more appropriate than word counts, especially in the case of Chinese, where there is substantial disagreement about the relationship between words and characters, since many multiple character combinations can be interpreted as single words or multiple words (Zydroń 2004).

Issues of word count are very contentious since they affect the bottom line of almost any localization-related task, and translators have typically resisted efforts to standardize word count that they fear might deprive them of justifiable earnings. Consequently, the approach of GMX is not to replace existing word count mechanisms, but rather to provide a standard comparative method for comparison. The choice of tools used to derive source word counts will continue to depend on the characteristics of the source text, since different linguistic tools address different needs. For example, a specific linguistic tool might include the capability to automatically translate dates and other well-defined textual data without human intervention, and produce an appropriate word count for translation tasks that does not include these items; another tool, on the other hand, might include these items in a word count. Each tool is appropriate for its given tasks, but their word count numbers would not be transferable. By basing quotes on GMX-V word counts however, price comparisons and tool performance can be made more transparent and better serve the business needs of companies in various situations and provide easier comparison of efficiency and reuse. Because GMX-V has not yet been adopted as a standard, it is difficult to forecast the exact purposes to which it will be put.

Discussion of the various standards initiatives being conducted at LISA underscores the importance and scope of standards initiatives within LISA and its OSCAR special interest group. When these standards are examined as a whole, it can be seen that they match the needs of businesses shifting to information-centered paradigms. Standards like TMX, SRX, and TBX help maintain information abstraction and coherence between multiple language versions of information. GMX serves a separate ancillary purpose: it facilitates the quantification of the costs associated with the production and maintenance of commoditized information.

OASIS standards

The Organization for the Advancement of Structured Information Standards (OASIS) is not a localization-related organization. Rather, it is involved in the standardization of business processes and information using XML. OASIS is the governing body for two additional standards of relevance to the localization and translation industries.

XLIFF — XML Localization Interchange File Format

XLIFF is the result of the cooperation of a number of OASIS-member companies, many of them LISA members as well, to create a format for the representation of localizable data. According to the XLIFF committee's charter (XLIFF 2005):

> The purpose of the OASIS XLIFF TC is to define, through XML vocabularies, an extensible specification for the interchange of localization information. The specification will provide the ability to mark up and capture localizable data and interoperate with different processes or phases without loss of information. The vocabularies will be tool-neutral, support the localization-related aspects of internationalization and the entire localization process. The vocabularies will support common software and content data formats. The specification will provide an extensibility mechanism to allow the development of tools compatible with an implementer's own proprietary data formats and workflow requirements.

XLIFF is commonly used as a way to bundle localizable elements and send them to localization service providers. In addition, at least one company (Novell) has been reported to use XLIFF as its default internal resource format for software (Cattin de Bois 2004).

The XLIFF and TMX committees share a number of members, and the design philosophies of the two standards show certain parallels, even though they are used for different purposes. Like TMX, XLIFF facilitates the abstraction of information from a language-specific instantiation into a multilingual environment, but is not used to populate a TM database. However, XLIFF files can be leveraged by recycling previous translations from TMX files or TM databases. The two standards thus address different facets of the data abstraction puzzle.

Trans-WS (Translation Web Services)

Trans-WS is not yet an approved standard, but is well on its way to becoming one, with a draft specification nearly complete. Trans-WS is a different sort of standard than those discussed previously: it will establish a standard protocol for the transmission of localization products and associated metadata between companies using the Internet. Its primary function will be to simplify basic workflow issues and job requirements that are generic to most localization jobs (Reynolds 2004). The promise of Trans-WS is that issues of file transmission, job specification, communication between clients and localization service providers, and job tracking can be automated in a standard way to reduce the manual burden associated with managing localization jobs.

Conclusion

Contrary to popular belief, localization is not a product of the PC revolution that began in the 1980s: as shown, its roots go back much further. The PC revolution, however, did change the way in which information and knowledge relate to each other and to the goals of localization. The shift from a knowledge-centered model to an information-centered model of localization has had profound implications for the way in which information is treated within businesses. Because of the increasing abstraction of information away from specific instantiations to generalizable representations that can be reused, and the resultant commoditization of information, standards that facilitate the reuse of information in heterogeneous environments (and across linguistic and cultural boundaries) have risen in importance. These standards can have tremendous economic importance by allowing the value of information to persist over time and by liberating information from dependence upon products that do not themselves contribute significantly to the value of the information.

Standards cited in this document

Extensible Markup Language (XML). http://www.w3.org/XML/
GILT Metrics eXchange — Volume (GMX-V). http://www.lisa.org/oscar/gmx/
Segmentation Rules eXchange (SRX). http://www.lisa.org/oscar/seg/
TBX Link. http://www.lisa.org/oscar/tbxlink/
TermBase eXchange (TBX). http://www.lisa.org/tbx/
Translation Memory eXchange (TMX). http://www.lisa.org/tmx/
Translation Web Services (TransWS). http://www.lisa.org/oscar/tbxlink/
XML Localization Interchange File Format (XLIFF). http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xliff

References

Akoff, R.L. 1989. "From data to wisdom." *Journal of Applied Systems Analysis* 16: 3–9.
Berners-Lee, T., Hendler, J. and Lassila, O. 2001. "The semantic web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities." *Scientific American* 284 (5): 35–43.

Cattin de Bois, G. 2004. "OSCAR: Visioning the future of standards." *Globalization Insider* 13 (2.1). http://www.lisa.org/archive_domain/newsletters/2004/2.1/CattindesBois.html

Lommel, A. 2004. *LISA 2004 Translation Memory Survey: Translation Memory and Translation Memory Standards*. Romainmôtier, Switzerland: LISA. http://www.lisa.org/products/survey/2004/tmsurvey.html

——— . 2005. *Taking Software to the World: Results of the 2005 Global Software Survey*. Romainmôtier, Switzerland: LISA. https://www.lisa.org/products/gssurvey.html

Rau, F. 1997. "OSCAR: New SIG to develop translation memory exchange format standard. *LISA Newsletter* 6 (3). http://www.lisa.org/archive_domain/newsletters/1997/3/rau.html

Reynolds, P. 2004. "Web service for translation." *Globalization Insider* 13 (2.3). http://www.lisa.org/archive_domain/newsletters/2004/2.3/reynolds.html

SALT. 2003. "Standards-based access to multilingual lexicons and terminologies." *SALT Technical Website*. http://www.loria.fr/projets/SALT/saltsite.html

XLIFF. 2005. *Charter of the OASIS XML Localisation Interchange File Format TC*. http://www.oasis-open.org/committees/xliff/charter.php

Zydroń, A. 2004. "GILT metrics — Slaying the word count dragon." *Globalization Insider* 13 (4.1). http://www.lisa.org/archive_domain/newsletters/2004/4.1/zydron.html

The creation and application of language industry standards

Sue Ellen Wright

1. Introduction

This article provides an overview of the standards that are being used today in the language industry, with an emphasis on the localization arena. Viewed according to type, these standards can be broken down into standards related to content creation, to translation and localization, to terminology, to ontologies, and to locale specification, as well as basic standards, such as those that govern language and character codes. Standards specific to the localization field tend to be related directly to the management of terminology, legacy translations, or workflow and project management, but it is important to bear in mind that these standards are frequently based on a foundation of more generic standards that underpin technical writing, documentation, and the World Wide Web environment.

Another way to categorize language industry standards is to look at the various organizations involved in the standardization process. The names of the groups responsible for the standards, along with the standards themselves, comprise a proverbial alphabet soup of acronyms, abbreviations, and short-form terms. In order to facilitate a more direct presentation and to provide the reader with a ready reference to this information, an Appendix (see p. 333 of this volume) lists these acronyms, together with their full forms and brief descriptions, including Web links and bibliographic information on relevant standards. Titles are referenced in the form (including punctuation and capitalization) preferred by the individual bodies, and are collated according to standard number in order to ensure a systematic view of families of standards.

2. Industrial standards and the evolution of language standards

The US National Standards Policy Advisory Committee defines a standard as:

> A prescribed set of rules, conditions, or requirements concerning definitions of terms; classification of components; specification of materials, performance, or operations; delineation of procedures; or measurement of quantity and quality in describing materials, products, systems, services, or practices (Techstreet 2004).

With the possible exception of "materials," language-related standards address all these areas.

2.1 Industrial standards

Historians tend to document the origin of standards starting with the normalization of such things as the length of the cubit or the like, but public perception of standards appropriately focuses on the modern industrial sector, particularly with reference to the kinds of standards introduced in the wake of the industrial revolution, beginning in the early 19th and continuing through the 20th centuries to the present. The determination of uniform dimensions and properties in everything from fire hose fittings to pipe and screw sizes, from steel composition to condom performance, reflects standards activities involving:

- Weights and measures
- Safety and the protection of property (e.g., fire hose regulations)
- Product quality and consumer protection
- Mass production and component interchangeability
- Large scale commercial operations requiring the interoperability of many different types of products and services (e.g., in civil and construction engineering)
- Consistency across regional and international borders (e.g., standardized rail gauges) (ANSI 2005a)

Early standards dealt primarily with product specifications, augmented by process standards to assure uniform procedures. The stringency with which standards are specified varies on a modal scale from *can* to *should* to the ultimate prescriptive, *shall*. Validity and enforceability ranges from regulatory standards, which are codified in law, to voluntary standards, which are an expression of industry self-governance. In some countries (e.g., the former Soviet Union), all official standards, even language standards, have the force of law, at least in theory, while in other countries (e.g., the US) most industrial standards are voluntary, but adherence is very widespread due to high consumer expectations for product quality and safety, coupled with a litigious culture that imposes costly *de facto* punishment for non-compliance.

In addition to weights and measures, which remain a core concern, modern standards types include:

- Product standards
- Process and performance standards
- Guides (recommendations of uniform procedures)
- Functional standards that actually create or support technologies (e.g., TCP/IP, HTML, etc.)
- Terminology standards

2.2 Language standards

For many years, the only official language standards consisted of lists of standardized terminologies published by the technical committees of the major standards organizations. Drafters of industrial standards in effect “invented” terminology standards in order to assure the precise definition of the terms used in their specifications, a practice that began in the early years of the twentieth century on an international scale in ISA/ISO and nationally in ASTM, SAE, DIN, ÖNORM, etc., and gathered momentum in the decades following World War II (Felber 1984: 16 ff). Reflecting practices followed in laws and contracts, standardized terminology is based on the prenegotiation of meaning in order to avoid ambiguity and ensure stakeholder consensus.

The trend toward more comprehensive, and consequently more powerful, language standards is closely associated with the development of *language engineering* and *language technology* (see Wright 1998 and 2001) as a function of the digitization of language resources. Wright’s premise in 1998 was that classic engineering (e.g., mechanical or civil engineering) grew out of pure physics when the identification of individual units of information about the material world led to the quantification (and in modern times to the digitization) of knowledge about the constituents of engineering objects. In the computer age, the identification and markup of linguistic units and the ability to store and retrieve discrete elements of information about language and text has led to the development of a wide variety of human language technologies (HLT). Many of the resulting technologies have been applied to the translation and localization process.

2.3 Characterizing language standards

The Localisation Industry Standards Association (LISA) presents a good global view of localization-relevant standards (LISA 2004), including Cadieux’s *Automated Localization Workflow Model*, which superimposes standards references at critical nodes in his representation of the workflow cycle. Wright and McClure (2003) attempted a comprehensive overview of standards activities, and Wright maintains a retrospective Web site (Wright 2005) with sporadic updates. Of course, the best way to stay abreast with ongoing developments is to follow industry news sources such as *MultiLingual Computing & Technology*, OASIS's Web-based *Cover Pages*, and *The LISA Newsletter*, but maintaining a clear perspective on the advancing positions of the different standards remains a challenge. This problem is exacerbated by the lack of any centralized steering authority or clearinghouse where information about language standards efforts could be posted.

2.3.1 *Types of language standards*

Wright and McClure group language standards into seven major areas:

- *Base standards* include various markup languages (SGML, XML, HTML, etc.), together with metadata resources, and standards governing character coding, access protocols and data interoperability.
- *Content creation, manipulation, and maintenance* usually take the form of XML-based authoring standards and standards for text and content markup.
- *Translation standards* (which in general address localization issues as well as just translation) define parameters for negotiating translation and localization contracts (e.g., DIN 2345, ASTM 15.XXXX), provide metrics for evaluating translations (SAE J2450, ATA Framework, The LISA QA Model), facilitate the interchange and segmentation of translation memory units (LISA TMX and SRX), and specify procedures for certifying translation service providers (CEN EN 15038:2006, Translation services — Service requirements).
- *Terminology and lexicography standards*, specifically the standards of ISO TC 37 (Terminology and language and content resources), treat a range of topics, from terminology theory to data interchange. Controlled language standards prescribe more restrictive style, grammar, and terminology usage.
- *Taxonomy and ontology standards*, specifically ISO and ANSI thesaurus standards and the Web Ontology Language (OWL), together with the W3C's *Simple Knowledge Organization System (SKOS)* standard, are designed to support intelligent information retrieval and ultimately the evolution of the Semantic Web.
- *Corpus management standards* treat specific markup conventions for logical, syntactic, and semantic markup of text corpora.
- *Language and locale-related standards* are rapidly progressing to include all the world's languages (and possibly dialects as well), as well as extending to include a range of information specific to the implementation of localization strategies.

2.3.2 *Standards organizations*

Characterization of these standards by type implies a neatly organized field of activity, but in reality a proliferation of organizations and interest groups, both formal standards institutes and various industry consortia, contribute on an ongoing basis to the catalog of language industry standards. Not only is it difficult to maintain a clear view of all these different efforts — the work of different groups also sometimes tends to overlap or conflict, not infrequently resulting in contradictory or competitive standards.

Certainly the most established players in the field are ISO and the IEC, and their Joint Technical Committee, JTC 1, which is responsible for, among other things, character codes (SC 2, ISO/IEC 10646, ISO's mirror standard to Unicode), SGML (SC 34, ISO/IEC 8879), and metadata registries (SC 32, ISO/IEC 11179). National bodies associated with ISO/IEC maintain mirror advisory groups at national levels and also field their own specific standards (ANSI, DIN, ÖNORM, AFNOR, BSI, etc., together with the approximately 200 US specialized groups under the ANSI umbrella, e.g., ASTM, SAE; see ANSI 2005b).

Outside the ISO/IEC environment, special interest groups such as the IETF, IANA, and the W3C govern the Internet and the World Wide Web, respectively. Organizations such as LISA and OASIS are for the most part made up of companies active in the localization industry and in e-business, with both groups producing standards, guides, and metrics for use in localization environments. Some quasi-standards are not even fielded by normative organizations *per se*: the ATA Standard Framework for Error Marking, for instance, can be used to produce a viable metric for assessing translation quality, but it arose in response to the need to objectively assess performance in the context of translator certification.

2.3.3 *De facto standards*

Other standards are not promulgated by standards organizations or by professional groups. For instance, many functionalities of Microsoft software and the overall Microsoft environment amount to quasi-standards, especially the implementation of HTML as dictated by Microsoft Internet Explorer™. Microsoft locale encoding has in the past competed with normative practice, but in recent years Microsoft has become actively involved in a number of standards initiatives, particularly OASIS, Unicode, and the Language Codes. The Linux-related open standards movement (FSG) represents an effort to resist the market dominance of Microsoft, in effect embodying a set of counter-standards that are establishing themselves as an alternative in the marketplace.

It is not unusual for a standard to evolve out of one group (such as the FSG) and move into a broader-based, more prestigious venue, such as the W3C or Unicode. This pattern describes the trajectory of Unicode's *Locale Data Markup Language*. In similar fashion, independent and company-based de facto standards not infrequently migrate into the ISO/IEC world, such as has been the case with SGML, Unicode, and more recently, SIL International's *Ethnologue* resource, which is becoming the basis for an expanded ISO 639. These kinds of standards grow out of real, core enterprise needs and reflect a concentration of research and development investment within company or organizational contexts that would in many cases be out of the reach of volunteer organizations like ISO. For instance, SIL's commitment to developing educational and public service resources for speakers of languages that previously had no written form has lead to the systematic codification of all the world's languages within a linguistically oriented framework (as opposed, for instance, to the library classification structure of the current ISO 639-2). Furthermore, the fact that this resource has been openly published on the Web for many years means that it has already been rigorously vetted by users all over the world, which qualifies it as a mature candidate for standardization.

2.3.4 *Standards philosophies*

The availability and accessibility of standards varies considerably, depending on the source of the standard in question and the marketing and distribution philosophy adopted by each sponsoring group. It is often said that ISO/IEC and their national affiliates tend to function as publishing houses masquerading as standards organizations. DIN, for instance, maintains a nice distinction between the standards institute and its publishing affiliate, Beuth Verlag, but the internal administration of the standards organizations is still for the most part financed by the sale of standards, and the copyright for these standards is very closely held. (Actual creation of the standards is contributed by experts who are generally financed by industry and government.)

Exceptions to jealous copyright protection include widely used standards such as the multi-section ISO 639 (Language Codes), ISO 3166 (Country Codes), and ISO 15924 (Scripts). In 2003, there was a brief controversy concerning the notion that the standards organizations might claim royalties for the right to incorporate these codes into software, but serious objections were voiced throughout the computing community, particularly from W3C, from companies like Microsoft, and from originators of the standards-related content such as SIL and ISO's own TC 37, which co-administers the language codes. The brouhaha abated when ISO reiterated its existing practice that important standards such as these codes would remain freely available for all types of use and for incorporation into software applications.

IETF and W3C, along with a number of other Web-based groups, freely publish their work on the Web and invite interested experts to participate in the creation of their standards. LISA makes many of its documents available in this way, with the exception of its QA Model, which is configured as a database application, and whose pricing is more in line with niche computer programs than with standards. The marketing of what is essentially a standard in the form of a functional application underscores the fact that printed standards (including the snapshot-style graphic PDF files marketed as "electronic versions" by the traditional standards organizations) are not particularly useful in computing environments. Functional standards in particular have more value and are more likely to be implemented if they are made available in processable electronic form. For instance, a data element standard listing a full metadata registry for a subject domain will be of little utility on paper, but can be accessed according to flexible search strategies, freely subsetted, and/or built directly into applications if the information is available as an xml-based data resource.

3. Weights and measures: The coding standards

In the arena of language standards, the character, language, script, and country codes are analogous to the baseline weights and measures that are the subject of standards governing industrial practice in the physical sciences. The following discussion addresses issues involving the history, expansion, and application of the standardized language identifiers. These include the ISO 639 family of *language codes*, the *language tag* (*lang)* used in SGML and HTML, the *xml:lang* attribute used in XML, and the closely related locale identifiers (*locale IDs*), which are based on language and country code elements.

3.1 Language codes

The specification of standard codes for languages and their use with the codes for countries grew out of practices in the terminology and library communities prior to the advent of widespread computerization and before there was any perceived mission to include all the world's languages in comprehensive multilingual information management resources. Original terminology management activities were limited to a smaller group of so-called major languages, most of them European, so the notoriously limited permutation provided by a two-letter code was sufficient for early terminologists, which resulted in the narrow set of languages covered by the ISO 639-1 *Alpha-2 language code*. Librarians, in contrast, were confronted with the need to catalog and retrieve documents in a wider range of languages, with the consequence that they created a three-letter code to provide broader, though hardly comprehensive, coverage.

With the computerization of library records in the last third of the twentieth century, huge volumes of legacy data were created in the United States using the Library of Congress (LOC) system (the MARC record), which adopted English-based codes for several of the major languages (*spa, fre, ger*), while European libraries and terminologists persisted in using codes reflecting the name of the languages in those languages (*esp, fra, deu*). When ISO finally issued a three-letter code (ISO 639-2:1998, the *Alpha-3 language code*), both options were retained, partly for political reasons, but essentially in order to protect the masses of legacy data residing in the two systems. The LOC English-based codes are commonly called "the library codes," and, although used widely by librarians elsewhere in the world, the *esp, fra, deu* style of codes was designated the "terminology codes," sometimes called the "localized codes." The computing community promptly protested the potential ambiguity posed by the presence of optional codes, but pragmatically came to terms with the issue by specifying the terminology codes for use in computing environments and deprecating the use of the library codes. In 2001, the Joint Advisory Committee for ISO 639 determined that the bibliography and terminology codes (B/T) would henceforth be listed as synonyms and not treated as alternatives (Cover 2005).

Aside from issues involving multiple forms and language groupings, the limitations of the current ISO 639-2:1998 in terms of language coverage and linguistic rigor have caused the standard to be overshadowed by SIL's *Ethnologue* collection, which provides three-letter codes for over 6000 languages. Although ISO's Joint Advisory Committee (JAC) administering the standard is moving to extend the current collection to cover additional codes upon request, the fundamental solution to this problem will come with the adoption of a new ISO 639-3, which includes the entire *Ethnologue* set, with some adjustments for group codes and anomalies. This process has taken time primarily because of the need to reconcile non-linguistic classification principles used in the 639-2, which make it difficult to equate all codes from the two sets. Speedy adoption is important, however, because there remains the danger that major stakeholders might simply develop or reinstate their own coding systems because the current system fails to meet their needs. The computing community is nearly unanimous, however, in supporting the view that inconsistent standards in the coding area would be counterproductive for everyone in the long run. (As of autumn 2005, this standard was moving swiftly to final adoption as an International Standard.)

The language code standards also specify a method for combining language and country codes in order to arrive at a more specific designation: *fr* is simply *French* as spoken or written anywhere in the world, whereas *fr-FR* is *French* as used in *France*. The current IETF 3066 sanctions this procedure and provides default rules for dealing with the potential ambiguities posed by the existence of both two and three letter alpha codes. IANA specifies this method for use on the Internet, and the W3C XML recommendation defines it as xml:lang, the official language identifier to be used in XML. Although case sensitivity is reiterated in 3066, the order of the elements and the number of characters (alpha 2 or 3) determine the semantics of the code, which has led in practice to the view that case is not necessarily a required distinguishing feature.

It should be noted that neither 639-1 nor 639-2 actually defines *what a language is*, nor what the codes listed in the standards actually designate. Efforts to define language abound, and one can perhaps assume that the understanding of these standards lies somewhere close to Crystal's distinction: "the abstract system underlying the spoken, written, or signed behaviour of a whole community" (2001: 184). Functionally, however, the two-letter codes as well as the three-letter terminology codes originally were used to identify the *object language* of data element content in terminological resources, whereas the library codes were used to identify the language associated with an archivable object such as a document or book. As such, the latter do not always adhere to Crystal's principle or to notions of mutual intelligibility, because some of the groupings employed (e.g., *Niger-Kordofanian (Other)*, for instance) serve rather like thesaurus *Use For* descriptors, employed to retrieve documents and information, but perhaps abhorrent to a speaker of one of these numerous languages, who feels no linguistic affinity to any language that might be called *Niger*. In contrast, the *Ethnologue* codes purport to be more strictly linguistic in approach, which leads to a stress on "inherent understanding … at a functional level" among *speakers* of an individual language (SIL 2005). Despite the emphasis on spoken language, *common literature* and *common ethnolinguistic identity* can serve to unite a set of mutually unintelligible dialects, such as Low German vs. Swiss German, and well-established distinct ethnolinguistic identities can provide grounds for splitting mutually intelligible dialects into separate language (e.g., the Scandinavian languages). This slightly hair-splitting distinction is still causing problems, however, in that Arabic-speaking nations are asserting that there is but one Arabic language, although speakers in different regions do not necessarily comprehend different regional variants. By analogy, one could convincingly argue that the same is true of German, where educated speakers can and do communicate easily with one another, although regional dialects can easily be mutually unintelligible.

3.2 Locale IDs

Constable and Simons (2000) have long contended that the combination code simply specifies a language, nothing more and nothing less. As such, it is frequently used in this way to identify both working and object languages in text, help, and interface files, following familiar conventions for cascading inheritance. Nevertheless, the combined code is also widely used to identify locales associated with the projected user of a computer program, or in other words, a service provided by or in conjunction with the program: "In general terms, the locale id is a parameter that is supplied to a particular service (date formatting, sorting, spell-checking, etc.)" (Davis 2004). This "service" can involve the language a user will want to see on screen and a set of cultural and administrative conventions associated with the geographic region in question, such as spelling variants, default separators, decimal number markers, punctuation guidelines, date/time format, currency symbols, etc. Davis notes in the draft Unicode UTS #35 *Locale Data Markup Language* that the language identifying function originally assigned to the language codes has been widely conflated with respect to the compound language codes and the locale IDs. He also asserts that, in contrast to ISO 639, where "only spoken language matters," so far as locales used in computing environments are concerned, "two languages are different if they require substantially different localized resources" (Davis 2004: 118). Despite this emphasis placed on distinctions based on spoken language, the ISO 639 codes, even in the *Ethnologue* sub-set, are probably more widely used for written materials than for any other medium, a position supported by the Library of Congress's Rebecca Guenther: "the existing 639-2 list (and ISO 639-1) has been developed for use with written languages" (Cover 2005). Hence it could be argued that the real distinguishing issue between the original application of the various *language codes* and their reincarnation as *locale IDs* is not so much whether they are based on spoken or written language, but rather, whether they are applied to language as employed in *static* and *dynamic* environments. The traditional application of terminology and library codes is static: they are permanent attributes of either real or data objects. In contrast, when ideally utilized, locale IDs can be used to switch functionalities on the fly, such as with the Windows™ Multilingual User Interface (MUI).

Not surprisingly, the needs of computer programmers, and in particular the needs for identifying languages and language regions in Internet and Web environments, are vastly more complex than the very limited specific needs for which the language coding system was originally designed. Cover (2004; 2005) attempts to provide a comprehensive overview of issues and accessible documents related to the definition of language identifiers in markup context, as well as other problems involving language codes, the details of which far exceed the scope of this article.

Essentially, the application of locales entails two critical considerations: what components will be incorporated into a locale code, and what its functionality will actually be within computing environments. The specification of language and local identification in computing environments takes several forms, specifically the *lang* attribute used traditionally in SGML and HTML, the *xml:lang* attribute use in, obviously, XML, and the *locale IDs* used to control various programming functions.

Phillips and Davis's draft (2004) for a revised IETF 3066 (IETF FRC 3066bis) proposes an expanded syntax for language tags used in Internet documents, the most critical subtags of which can include the language code (2 or 3 letter *lang*), a four-letter script tag (ISO 15924), a region code (2 or 3 letter country code) and/or a variant identifier. An example from the proposed standard might be: en-Latn-US-boont, with the understanding that in this case the four-letter script code *Latn* is probably superfluous and would usually be omitted because English is not normally represented in any other script. The final element represents an IANA-registered American English dialect. The standard clearly spells out conditions for such registrations. TC 37/SC 4 is currently discussing other refinements that would provide for reporting orthographic variations such as those encountered in Japanese.

The actual information referenced by such a locale ID is not, however, limited to the semantics of the identifier's components. IEEE's POSIX standard, whose primary focus is on interoperability between operating systems, provides for locales to reference character classification and case conversion; collation order; monetary formatting; numeric, non-monetary formatting; date and time formats; and message formatting. The ICU's *Locale Explorer* (2004) spells out standard information for ninety-one different languages, further divided into one hundred and eighty-two regions and variants. The FSG/open18n groups initially proposed a standardized extensible solution to locale markup in XML environments called the *Locale Data Markup Language*, which since December 2004 has been incorporated into the Unicode context in the form of a Unicode Technical Standard, UTS #35. The Unicode Consortium has also taken over from the FSG/open18n the Common Locale Data Repository (CLDR), whose purpose "is to provide a general XML format for the exchange of locale information for use in application and system development, and to gather, store, and make available a *common* set of locale data generated in that format" (CLDR 2005).

3.3 Character codes

A discussion of standard codes would not be complete without addressing the adoption of Unicode character encoding. Unicode provides unique identifiers (code points) for each character in every one of the world's major languages (and most minor ones) written today, as well as a number of archaic languages that are nonetheless the subject of scholarly inquiry. Not all archaic languages are included, however, but specialists world-wide continue to create the encoding and character representations for previously excluded languages. The most recent version of Unicode, Version 4.0, provides codes for 96,447 characters from the world's alphabets, ideograph sets, and symbol collections (Unicode 2005), including new codes for traditional Chinese, for instance. The primary purpose of the standard has been to eliminate conflicting processing issues that have arisen as the result of a proliferation of encoding systems according to platform issues, language treatment in different parts of the world, and difficulties arising from variation in character handling among languages.

> *Unicode provides a unique number for every character,*
> *no matter what the platform,*
> *no matter what the program,*
> *no matter what the language* (Unicode 2005).

While the universal implementation of Unicode sounds like the answer to text processing in multilingual environments, it should not be assumed that universal adoption will be easy or immediate. Even given the best of conditions, the representation of any given script depends upon the presence of the required fonts and the appropriate configuration of operating systems, programs, and Web browsers. A simple trip to the Unicode Web site will provide some notion of a system's capabilities — my system, for instance, does not display Amharic, Blin, or Tigrigna, but it handles East Asian characters quite nicely because I've configured it to do that. Data input devices (keyboards in particular) also need to be configured to facilitate data entry in a given language. The Web site provides information on how to configure one's computer to display missing languages.

Issues involving character encoding go beyond the mere representation of characters to include their behaviors in text, behaviors which in many cases are complex in comparison to the Latin, Greek or Cyrillic familiar in European languages. Many other issues, such as handling bidirectional text and ruby text in Japanese have already been addressed, but the solutions for other "problems" are still the subject of discussion, such as the normalization (loss-less roundtrip) of legacy character encodings embedded in Unicode texts.

In addition to resolving such issues, encouraging the adoption of the Unicode solution is probably the main focus of "Unicadets" worldwide. As will be seen later with respect to functional standards (TMX in particular), market forces can be brought to bear on adoption policies. Localizers in particular can insist to product vendors that they implement Unicode in all new products and product upgrades and that Web designers account for Unicode in their designs.

4. Standards for quality control and quality assurance

State-of-the-art language industry standards mirror best practices in industrial standards. Reflecting a transition that has been widely completed in the manufacturing and services sector, language-related QA standards are undergoing an evolution from the notion of quality control to that of quality assurance. Within this context, translation and localization quality are judged on the basis of text "quality," however it may be defined for the localization of mostly commercial products. The distinction between *process* and *product*, which has been a part of theoretical debate in translation studies for decades, aligns serendipitously with the process/product approach of the core ISO 9000 quality system. Finally, this article will consider current industry efforts to establish process standards and product metrics, such as the DIN, ÖNORM and ASTM guidelines governing contract negotiations for translation and localization services, and the SAE metric for translation quality.

4.1 Quality control vs. quality assurance

ISO 9000:2000 states that *quality assurance* (QA) provides confidence that quality requirements will be fulfilled, whereas *quality control* (QC) focuses on checking quality product and services. QA encompasses procedures set in place to assure and document quality throughout the production process, whereas QC in the localization industry has historically involved *end-item inspection*, in the form of editing, proofreading, and product testing. The LISA QA Model sums up its QA orientation as "an ongoing process [that] should not happen just at the end of a project" (LISA QA Model 2004: 7).

Although this distinction between QA and QC is perhaps more honored in the breach than in actual practice, it reflects historical development from QC via QA to the current concept of *total quality management* (TQM) and provides the structure for the standards hierarchy described in 4.4.1.

4.2 Typology for pragmatic translation in localization environments

Schnitzlein (2003; 2 ff.) observes that there is all too often a disconnect between practice, research (pragmatic translation studies), and training, not to mention a further disconnect between approaches used in industry and government. There is actually good reason for industry's relative disregard for translation theory insofar as theory has focused solely on literary values or tried to transfer sometimes obtuse philosophical concerns to discussions on pragmatic translation. Truth to tell, much of this effort is unproductive with respect to localization. Office workers sitting down to use a new word processing software in Sri Lanka or Singapore are not interested in perceiving the socio-cultural essence of the source text shining through a "foreignized" translation — they only want to get on with their tasks, using their own languages and counting on the program to support them with the appropriate character sets, currency values, formatting conventions, etc. Regardless which of the more or less "standard" programs they use, the programming interface itself comprises its own cultural environment, which should at this point be roughly international, stripped of any references to source language or culture. Nevertheless, there are some theoretical views that are useful for evaluating quality in localization environments.

4.2.1 *Equivalence and the translation brief*

Translation quality has generally been defined in terms of equivalence. In light of age-old controversies surrounding what constitutes textual equivalence (as opposed to terminological equivalence, which is much easier to characterize), one trend in translation theory equates equivalence to *adequacy* with respect to a *translation brief*, which for purposes of this article can be defined as the detailed specification of requirements for a given language mediation task (Nord 1997). Such detailed specifications can be set down in an agreement for translation or localization services as described in the draft ASTM F15.XXXX standard (ASTM 2005); Melby and Manning have recently developed the relevance of the contribution that Nord's approach has made to the standard, in the context of various perspectives on meaning (Melby et al., forthcoming).

4.2.2 *Translation typology*

House posits two fundamental types of equivalence, resulting in two types of translation: overt translation and covert translation (House 1997: 66). In brief, covert translation results in a target text that creates a second original that is totally fluent in the target language and substantially devoid of source-culture (source-locale) and textual markers that would reveal it as a translation. In contrast, overt translation retains these cultural markers and is understood by the reader to be a translation. Covert translation is usually the appropriate approach for the kind of pragmatic texts encountered in localization, whereas overt translation is generally associated with the translation of literary and culturally significant materials, such as many texts that are translated in national security and diplomatic venues. Variations on this theme come into play with the localization of computer games, where marketability issues in a given locale (e.g., the marketing of American or European games in a Japanese environment) compete with literary, fantasy aspects of a product in a push-pull synergy that requires its own unique approach to the overt/covert dichotomy (see Dietz 122–124 in this volume, and Carless 2004).

4.3 Traditions in translation assessment: Process vs. product

The standards are not alone in stressing process vs. product. In his evaluation of translation-related assessment approaches, Schnitzlein distinguishes between process quality and product quality in the evaluation of translation (2003: 4):

> Process quality in the translation industry refers to the quality of the processes and operations that take places from the release or acceptance of a translation order to the delivery of the requested product. In contrast, product quality comprises the quality of the translation product, which can further be differentiated in terms of formal product quality and linguistic product quality. (Translation SEW.)[1]

Neubert and Shreve (1992) also make a strong case for considering process separate from product. Analysis of the means of production (primarily translators, along with their support tools infrastructure) and of process quality contributes to a quality assurance view, with an emphasis on process control and continual improvement.

As demonstrated in 4.4.1 below, current industry standards focus on process and product assessment and do not address translator assessment *per se*. Nevertheless, translators are indeed assessed both before they are hired and on an ongoing basis during the production process. In industry, assessment is primarily *summative* in that it focuses on product conformance to stated criteria, but it can also be *formative*, provided that adequate feedback is provided to translators during the post-mortem project stage (Dunne 2004). A recent trend in translation assessment favors a distinction between *translation quality assessment* (i.e., evaluation of translation product) and *translation skills evaluation* (which uses product as an indicator of cognitive skills and translator competence; Colina 2003: 130). Such a two-pronged approach to translator assessment runs parallel to a distinction between product assessment and process control in that translators, tools, and workflow procedures contribute to the translation process. Although it can be in the best interests of a translation service provider to supply competent translators with feedback in order to nurture expertise (Shreve 2002), in practice, editors frequently only provide their perceived "correct" solutions for problem points so that corrected translations can move on through the document production chain as efficiently as possible (Klaudy 1995: 201–202). Often there is no explanation of changes, and there is a tendency to change text simply because there are other options, which can leave translators confused or resentful rather than improving their skills. Feedback and subcontractor evaluation unfortunately tend to fall by the wayside in today's fragmented, decentralized localization environments.

4.4 Industry standards and the language sector

4.4.1 *ISO 9000 and related practices*

The three-part ISO 9000:2000 family of standards encompasses a multi-level hierarchy of increasingly detailed and customizable standardized environments, frameworks, and specific applications (see Figure 1). ISO 9001 establishes a systematic, internationally recognized set of guidelines and requirements that provides a uniform global framework for TQM systems in industry and government. The standards are generically process-oriented and non-industry specific. They focus on "customer" requirements and define quality in terms of an enterprise's ability to meet those requirements. This definition of quality as a function of clearly articulated customer requirements mirrors the notion of "adequacy" with respect to a translation brief, which naturally suggests an effort to align the assessment of translation quality with the philosophy of ISO 9000.

As noted in Wright (1991), other features of the ISO 9000 approach are very relevant to typical translation environments. End-item final editing is augmented by careful process control at all steps in a project. Controlled, capable processes are designed to catch non-conformant product early in the (here document) production chain in order to eliminate the need for later rework, to increase efficiency, and to decrease the risk of downstream consequences due to the proliferation of discrepant product (e.g., misunderstandings and even critical accidents or other calamities due to translation error). Although risk analysis and prevention are perhaps more difficult in this environment than in common manufacturing practice, the introduction of a structured feedback loop is, as noted above, an important factor in decreasing future risk of "failure" in conducting translation and localization projects.

4.4.2 *QA approaches: FMEA*

FMEA Critical Items Lists: Parallel to the ISO 9000 standards there are a number of best practices that are frequently built into the individual quality procedures defined under the standard, depending on individual needs. One of these methodologies is *Failure Mode and Effect Analysis* (FMEA), a practice implemented during the design or planning stage of a product or service. FMEA provides a systematic procedure for identifying possible failures of a product or service, methods for determining the probability of such a failure, and procedures for tracing and eliminating causes. Product or service characteristics are typically listed and categorized in terms of criticality (catastrophic, critical, marginal, minor; see Juran and Gryna 1988: 13.28 ff.). The creation of a relevant Critical Items List (CIL) is a typical activity for any serious assessment schema.

Shreve also cites the need to define a "set of quality characteristics" in order to specify a performance model for evaluating translation (2002: 153). Schnitzlein reports that all the product-oriented standards he examined enumerate translation product characteristics, but the selection of items included tends to vary. It is also typical in language metrics to classify errors according to severity: generally (but not always): critical, serious (major), and minor. The implications of this assertion for localization, both including and as apart from translation, would be that there is a need to formulate a number of localization project "typologies" (similar to a

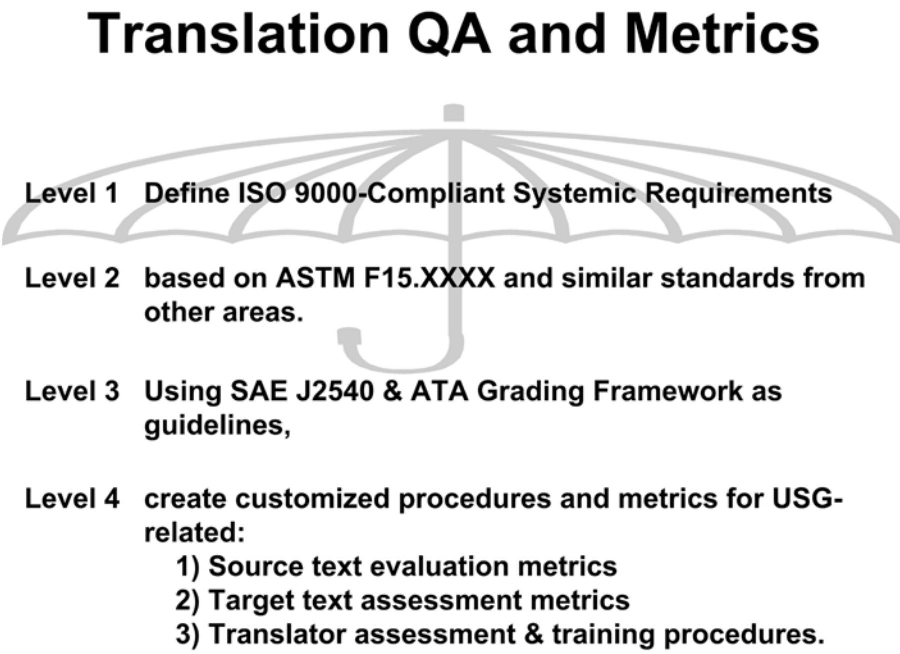

Figure 1. The standards cascade: A global industrial model.

variety of potential "translation briefs") for which it could be possible to describe stock sets of quality items to be controlled and evaluated. The LISA QA Model provides a beginning in this direction and should be a major focus of the evolving metrics effort outlined in 4.5.

Vinay and Darbelnet (1995), in their classic contrastive evaluation of English and French translation features, provide a compendium of important criteria that can be used to describe both translation product and the translation process, an approach that is more recently mirrored in Delisle et al. (1999), who distill the most useful terms from Vinay and Darbelnet, augment them with other critical terms used in international translation studies, and define these characteristics in four languages, primarily for use in translation pedagogy. Given the relative youth of localization activity, there is no such classic set of canonized criteria for evaluating localization as a separate activity. The industry needs to cull this kind of information from various sources, e.g., project planning and management milestones, post mortem evaluation points, etc.

4.4.3 *QA approaches: Repeatability and reproducibility (R&R)*

ISO 9000 and the entire FMEA exercise are intended to assure system capability, i.e., the ability of a *system* (as opposed to just individuals making up the system) to consistently produce product that conforms to TQM specifications. *Repeatability* represents the capability of a single operator using the same device to obtain comparable results through successive measurements, whereas *reproducibility* comprises the capability of multiple operators to use the same tool(s) over time and still achieve comparable results (Wright 1991).

Repeatability and reproducibility are critical to system capability because the level of quality, once achieved, must be replicated over time, regardless of who performs the task in question, but also because the task should ideally be executed with improved quality over time. Repeatability on the part of an individual translator is hampered by problems inherent in human translator memory. Reproducibility is obviously problematic because it is virtually impossible for a second translator to know material that someone else has processed.

Given the trend towards outsourcing and the fact that in many cases different translators/localizers/language engineers work on projects over time, the notion of reproducibility in particular poses serious challenges. These drawbacks are countered in modern translation environments by providing online access to information resources, parallel texts and corpora, translation memory, and terminology databases, all of which reflect an enterprise knowledge base, maintained either on the part of the original client or by service providers. Black-box mentality on the part of many commissioners of localization and translation services militates against the sharing of vital enterprise-specific knowledge, thus seriously impeding the implementation of systematic QA practices. Furthermore, even in cases where clients may be willing to share information, few companies today maintain the kind of global information resources that would be needed in order to assure desirable reproducibility.

4.4.4 *Process standards for the language industry*

At the next level in the ISO 9000 hierarchy, individual industries define their own more detailed process standards tailored to their requirements, procedures, and related best practices, such as industry-specific standards for the automotive, petroleum, and steel industries. The German and Austrian Standards Institutes (DIN 2345; ON D 1200 and D 1201) have both created standards defining procedures for contracts for translation products and services, laying down guidelines for specifying formal translation briefs for individual translation (or localization) projects or clients. In contrast to the other standards of this nature, ON D 1200 provides breakout criteria for levels of quality (e.g., *printable, standard, and working versions* of documents), in effect setting tolerance ranges for individual translation briefs. The ON standards also provide for standards bodies to serve as auditing agents.

The American Society for Testing and Materials (ASTM) has created consumer-protection standards for interpreting (interpretation) and translation (ASTM F2089-01 and F15.XXXX, a draft, as yet unnumbered, standard being elaborated by ASTM Technical Committee F15.48) that sets down documentable procedures (in the sense of ISO 9000 documentation criteria) for entering into contracts for language mediation services. It is not the purpose of these standards to prescribe what has to be in a translation or localization contract, but rather to suggest the important issues that need to be taken into consideration. Although such contracts will almost certainly specify the need for quality assurance procedures and evaluations, they do not dictate specific criteria for judging the actual translation product. An implicit assumption is that the parties to such contracts will identify specific metrics that are appropriate for their individual applications. A comparable CEN project, CEN BTTF 138 project is currently nearing completion and will provide criteria for setting up a certification system for corporate translation service providers.

4.4.5 *Translation quality metrics*

The next level in the hierarchical cascade of quality standards is occupied by "quantitative quality measurement procedures" (Schnitzlein 2003: 47), leading to the generation of so-called "quality metrics." Metric-related expressions of translation adequacy yield formal quantifiable numerical values by providing lists of critical characteristics, which are weighted numerically in order to objectify any errors that may be present. Two accessible approaches include the SAE metric specifically for "automotive service information" (SAE J 2540: 2001) and the American Translators Association (ATA) framework for grading the ATA certification examination. Ongoing projects in LISA and in the Localization Institute address the quantification of source text/content scope and quality, target product quality, the localization process, and standard productivity factors, all of these activities being categorized by their work groups as "metrics."

4.4.5.1 *The ILR Scale.* The original Interagency Language Roundtable (ILR) Language Proficiency Skill Level Descriptions provided a scale of 1–5 for evaluating foreign language professionals working for the federal government — NATO offers its own variant. The descriptions generally relate individual skill levels on the part of language professionals to specified tasks and so-called text difficulty levels. Performance criteria are listed in considerable detail for a progression of linguistic skills related to speaking, listening, reading, and writing, but these guidelines do not address any predictive relations between the ILR language skills ratings and translation-related performance skills. Indeed, the kinds of linguistic competence described by the ILR rating scale have comprised only a baseline on which actual language mediation and localization-related competences could be built. Thus the ILR scale has not been directly relevant to the needs of industry (or to government actually, for that matter). In order to meet this criticism, the ILR has developed provisional descriptions for translation skill levels, which start at very low levels for simple content recognition and progress to the ability to handle complex linguistic and cultural content in a variety of special subject fields. Within this broad range, only the top two levels are probably acceptable in the private sector, but it should be noted that the high volume of work required in government and national security environments coupled with the sometimes uncommon language combinations demanded in these areas results in the need to use some language professionals with otherwise marginal skills to perform triage and to route critical documents for translation by more competent translators.

4.4.5.2 *SAE J 2450:2001. SAE J2450* is an SAE Recommended Practice whose scope is to develop a metric for the evaluation of translation quality for service documentation in the automotive industry, regardless of the source or the target language, and regardless of how the translation is performed, i.e., human or machine translation. The standard identifies seven critical categories (see Table 1), each rated in terms of severity on an individually specified scale from 5 (severe) to 1 (trivial). While the SAE standard is important as a product quality metric, it does not address style, intertextuality, cultural significance, etc., all of which are critical in localization. The standard can be used to evaluate translation product in an industrial setting in order to select and audit translation providers. It can, for instance, be used to rate translation suppliers in order to cut cost and effort by replacing 100% inspection with supposedly "reliable" random sample editing checks.

Table 1. SAE J 2450: Error categories, classifications, and weights.

Category Name: (abbreviation)	**Sub-Classification: (abbreviation)**	**Weight: serious/minor**
a. Wrong Term (WT	serious (s)	5/2
b. Syntactic Error (SE)	minor (m)	4/2
c. Omission (OM)		4/2
d. Word Structure or Agreement Error (SA)		4/2
e. Misspelling (SP)		3/1
f. Punctuation Error (PE)		
g. Miscellaneous Error (ME)		3/1

4.4.5.3 *The ATA Framework for Standard Error Marking.* The ATA Framework (see Table 2) was developed by the American Translators Association Certification Committee in order to improve consistency in the ATA certification process and to render the system more transparent (ATA 2004). The exam is designed as a summative measure of translation expertise. Nevertheless, ATA grading practices are being adopted and adapted by academics for use in formative situations (see

Doyle 2003, as well as Baer and Koby 2003). The ATA CIL includes 22 criteria and five weighting levels, providing a far more comprehensive evaluative tool than the SAE metric, but this higher level of complexity reflects the fact that it is designed for assessing a full range of text types and subject matters.

4.4.5.4 The LISA QA Model is the most complex (and potentially useful) metric for measuring not only translation quality, but also all aspects of the localization process as well. The program, with its accompanying documentation, is now available in Version 3.0. Having recognized the limited value that many paper-oriented standards have in real operational environments, the designers of the metric have, as noted above, configured it as a stand-alone, but integratable, database application for purposes of automating the QA checking and decision-making process.

The LISA model documents a range of critical characteristics that are specific to software products, such as country or locale related items; issues involving document-related components, such as tables of contents, indexes, etc.; layout, sorting rules, typography and character encoding, which are particularly important for non-Latin programming environments; graphics call-outs and captions; and output functions. It provides for end-item inspection criteria having to do with delivery and printing, and views the product in the broader context of documentation functions with respect to terminological usage, GUI and functional consistency, as well as art and design issues.

Like SAE J2450 and the ATA Framework, the LISA QA Model includes some critical characteristics, designed to measure translation quality itself (e.g., grammar, semantics, punctuation, spelling, general style, register, language variants, as well as mistranslation, accuracy, and terminology). The list is certainly more exhaustive than the SAE list, which does not concern itself with elements of style and register, but is neither as complete nor as theoretical as the ATA catalog. It should not be assumed, however, that having an exhaustive list is necessarily a positive factor. The set of criteria selected for any metric should adhere to the *"necessary and sufficient"* rule in order to assure adequate checking without incurring excessive cost and effort.

Like SAE J2450, the LISA model follows the standard FMEA approach by weighting severity levels in terms of minor, major, and critical errors, although the software also provides for the definition of multiple severity levels according to the needs of a project. As with metrics in general, the stated purpose of the method is to provide a set of formal quantifiable values for assessing localization services, including both translation and formal quality, as well as other project-oriented performance factors.

Table 2. Modified ATA Framework for Error Marking.

Error Type (Minus Points)	**Code**	**1 pt**	**2 pts**	**4 pts**	**8 pts**	**16 pts**
Incomplete passage	INC					
Misunderstanding of the source text	MU					
Mistranslation into target language	MT					
• Addition	A					
• Omission	O					
• Overtranslation	OV					
• Terminology, word choice	T					
• Register	R					
• Too freely translated	F					
• Too literal, word-for-word	L					
• False cognate	FC					
• Indecision, gave more than one option	IND					
• Inconsistency	INC					
• Ambiguity	AMB					
Grammar	G					
Agreement	AGR					
Syntax (phrase/clause/sentence structure)	SY					
Punctuation	P					
Spelling	SP					
Accents and diacritics	AC					
Case (upper/lower)	C					
Word form (singular/plural, inflections, etc.)	WF					
Usage	U					
Mot juste	MJ					
Collocation	COL					
Style	ST					
Word order	WO					
Denominalization	DN					
Error Point Subtotals (–)						
Quality Points (+)						
Quality Points Subtotals (+)						
Explanation						
Total Error Points (–)						

4.5 LISA GILT metrics[2]

4.5.1 *GMX-V*

Although the LISA QA Model has been categorized here as a metric, it offers a broader overall quality view of project quality, with, for instance, translation quality characteristics spread over several different major headings. It does not provide the same kind of focused metric seen in the SAE J2450. A new work item (July 2004) proposed for the LISA OSCAR group called simply the *GILT Metrics 1.0 Specification* proposes a markup vocabulary for the unambiguous sizing and billing of GILT tasks. Dubbed GMX-V (GILT Metrics-Volume), this initial version of a first part of the standard addresses translation volume. The goal is to establish word and character counts for actual translatable content, as well as count categories that are not electronically verifiable, such as the number of screen shots required to finish a job, or certain kinds of formatting tasks. The draft standard asserts a clear interdependence with other standards, such as XLIFF, Unicode, TWS, and LISA's own SRX segmentation standard.

The image of the GILT Metric as one component among a set of interactive standards underscores the growing complexity, but also the growing power, of the interoperable XML-based markup formats. The draft addresses issues such as word boundaries, script-related issues, inline elements and linking mechanisms. It proposes that the metric be declared as a registered namespace so that it can be embedded in a variety of documents.

GMX-V is planned as one of three LISA metric standards, including GMX-C for complexity and GMX-Q for Quality.

4.5.2 *GMX-C*

The second part of the metric will provide a method for quantifying the potential complexity or difficulty of the source files being evaluated. So far the convener visualizes criteria such as topic, special terminology, special instructions, relative incidence of in-line elements, and constraints. This component of the package represents a new view with respect to providing a quantifiable score for assessing source texts. Currently service providers do indeed scale projects in terms of overall difficulty, but their criteria for doing so tend to be at least individualistic, if not outright subjective.

Assuming highly qualified translation and localization staff, the potential sophistication of ideas and presentation of the topic treated in a particular project may not significantly increase the difficulty of translation or the time needed to do a project, unless it becomes difficult to find qualified experts to work on the job. Using familiar criteria like the FOGG index is not always the most reliable predictor of translation difficulty: a well-reasoned, but highly specialized text will be readily understood and comfortably (if not easily) translated by the right competent translator. Ambiguous, poorly written source text or poorly internationalized files, on the contrary, may prove very troublesome, even if the topic or text involved is not in itself "difficult."

The kind of source text metric described here would not only contribute to the calculation of project bids and billings; it would also facilitate the assignment of qualified translators and localizers to participate in projects and justify using more expensive individuals with special skills where warranted. Articulation of internationalization metrics, together with their inclusion in project quotes and agreements, would facilitate negotiations with clients when attempting to "clean up" source files, and promote client education when working with long-term clients.

4.5.3 *GMX-Q*

The third part of the metric would provide a translation quality metric for use in LISA, perhaps along the lines of the SAE J2450 and the ATA framework. This intent underscores the need to establish such metrics for different sectors of the translation and localization market and sub-tasks within the process, based on specific needs and resources.

4.6 Prospects for applying QA practice and theory

4.6.1 *Inter-rater reliability*

Both US and European members of the SAE task force have tested SAE J2450. Results indicate that although the provision of a coherent CIL introduces a level of objectivity to the otherwise strongly subjective process of evaluating translation quality, there remain significant variations between individual evaluators, yielding undesirably low levels of inter-rater reliability. The ATA Certification Committee attempts to counteract low inter-rater reliability with grader training and collaboration. Its system is further facilitated by the fact that the graders deal with only a small set of passages every year, and yet there remain many questions about uniformity among the graders. These experiences would indicate that even when efforts are made to implement quantifiable evaluation procedures, translation and localization "quality" remains a highly subjective notion. SAE testers report that there are even differences of opinion on what constitutes a spelling error! If we see problems with inter-rater reliability with a simple metric like SAE's, one can only wonder how reliable the more complex rating systems can hope to be. Perhaps the only realistic goal is to aim for consistent *intra-rater reliability*, i.e., reliable, predictable performance from individual single raters over time.

4.6.2 *Reproducibility and decentralized sourcing*

Open issues involving the evaluation of localization quality, inter-rater reliability (or more specifically, un-reliability) and the questionable prospects for ensuring repeatability and reproducibility are particularly sobering in light of the current trend in business to decentralize the entire localization process. The prospect for gaining a firm grasp on quality issues throughout the subcontracting chain from the internationalization stage through final localization hand-off seems dim indeed if the processes involved are constantly passed on to a variety of untried partners. Ben Martin's insourcing approach from the old J.D. Edwards organization, with its meticulously internationalized single-source, multilingual, multi-product output model, stands out as a viable alternative, but one that "current wisdom" is highly unlikely to embrace, especially now that J.D. Edwards, with its exemplary in-house documentation department, has been sucked under in the corporate takeover wars (Martin 2001). Although the maintenance of in-house capabilities is arguably better suited to optimal quality, it is probably more realistic in today's business climate to at least promote stable, long-term, more direct relationships with proven, reliable vendors (and by association, with the vendor's subcontractors). Furthermore, a strong argument can be made for maintaining centralized control of certain key quality-related processes, such as responsibility for high-quality source content, for terminology management, and for the maintenance of high-quality translation memory databases. If small to medium-sized enterprises cannot afford to retain an in-house localization specialist charged with maintaining such resources, then it would be wise to ensure that these tasks are taken over by a trusted, long-term vendor.

4.7 The Localization Institute's Business and Productivity Metrics Initiatives

Apart from the notion of quality metrics and reflecting yet another common use of the word "metric," in the fall of 2004 the Localization Institute initiated an effort to conduct a pair of collaborative surveys designed to establish benchmark metrics for business costs and productivity in the localization industry. The intention was for these projects to address the lack of hard data in a business sector that has been broadly neglected from the standpoint of economic analysis. More descriptive than prescriptive in its approach, this project was intended to produce reports that will be made available to those companies who choose to pay a reasonable fee to participate in the survey, which is actually not a bad solution: in order to view the results, companies have to contribute to the collection of the data. The business metrics initiative originally drew up a list of report values and set about to collect data on localization and internationalization headcount, in-house vs. outsourced costs, globalization cost, revenue, and return on investment. Unfortunately, however, not enough data was collected in the initial round to support the creation of a comprehensive report. Some progress was made on the development of the productivity metric, but again, the pressures of work in the industry seem to stand in the way of the data collection effort. At the time of publication, the organizers intend to continue the project, with the hope of eventually producing concrete reports that will be useful to the industry.

5. Functional standards: Data interchange and interoperability

Looking back at industrial paradigms, standards governing screw threads guarantee interchangeability, interoperability, and replaceability of screws, but it is entirely possible to manufacture and use screws that do not conform to standards, provided that there is no need to mate them with incompatible parts. (Nails would be a better example because they generally don't have to mate with anything.) Process standards generally are designed with the goal of ensuring compliance for product standards, although modern process and consumer protection standards also focus on services. In the language industry, process standards and product metrics serve to specify best practices, define procedures, and provide quantifiable assessment characteristics for evaluating existing practices and ensuring continuously improved quality.

Although product and process standards play a role in the language industry, the markup languages that comprise the base for language technology can be classified as functional standards, which differ significantly from product standards. Like the standards that govern the Internet and the Web, functional standards are created not for the measurement and evaluation of existing practice, but rather in order to implement wholly new technologies. For instance, radio or television transmission and the production of DVDs or MP3 files are impossible without adherence to the standards that govern these media. Such resources usually interact with some sort of enabling hardware, which conforms to its own standards. Conformance to these standard becomes the *raison d'être* of the "product" in question. In the language industry, these functional standards tend to provide new ways to manipulate and interchange information and content without disturbing layout and other markup or programming features embedded in localizable text. They began in the SGML world, but have quickly segued into the more flexible and rapidly burgeoning XML arena, and current developers are working hard to ensure interoperability of the standards themselves.

5.1 Translation Memory eXchange (TMX)

LISA OSCAR's Translation Memory eXchange (TMX) is designed as a vendor-neutral open standard for storing and exchanging translation memories created by computer-assisted translation (CAT) systems and other tools (SDL International 2004). The most obvious benefit of translation memories is the ability of translators and localizers to access and reuse chunks of text (segments) that have already been (presumably correctly) translated. This capability was exploited early on by individual translators in order to increase their productivity and to facilitate the processing of frequently updated versions of large files.

The fundamental purpose behind exchanging translation memory is that the localization process, especially in globally distributed environments, can involve many hand-offs from one subcontractor to another, and different vendor or locally developed products may lend themselves to different aspects of an overall job. Historically, the principles on which the standard are based go back to a time when language engineers working in individual localization shops cobbled together (or brilliantly generated) individual strategies for solving each complex data or content transfer problem that reared its ugly head. The adoption of global solutions to interchange challenges eliminates the need for such one-off solutions.

In today's content management climate, more and more enterprises have come to realize that the information contained in text corpora of all kinds, including translation memories, comprises a major component of enterprise capital, a factor that is all the more critical for software companies and other institutions that are based fundamentally on information resources. Being able to maintain and store such resources in a universal exchange format frees enterprises from being locked into single vendor formats and vastly enhances their ability to reuse data in novel ways. (For instance, term extraction from TMX files is generally far superior to term extraction from straight text files.)

TMX offers two levels of implementation, a Level 1 where the content of each segment (seg) is plain text only, and Level 2, where the standard also supports content markup so that it is possible to recreate the format of the original document based on the translated version expressed as a TMX file. Only those products that comply with Level 2 in independent certification testing are qualified to advertise the TMX quality logo. All components of the standard, including certification compliance and verification materials, are freely available downloadable from the Web. This approach, which mirrors the testing lab paradigm familiar with industry safety labs such as TÜV and UL, has enabled OSCAR to assume an aggressive position in "marketing" their free standard: the goal is to see the standard accepted by as many software vendors as possible. If the user public is convinced that compliance with the standard is a *sine qua non* for usability, vendors will fall in line. A growing number of software vendors have achieved TMX certification, thus earning the right to stamp their products with the TMX logo.

5.2 Segmentation Rules eXchange (SRX)

One problem that arises with a TM standard is the incompatibility of text segmentation practice. While it may appear desirable at first to establish a uniform segmentation practice (sentence segments, for instance), practice varies for legitimate reasons. Many localizers prefer to work with paragraph-size segments in order to avoid problems when it comes to re-ordering knowledge units and the presentation of information in the target text, but others prefer sentence-by-sentence matching. In still other situations (translating patents, for instance), it is more meaningful to work at the sub-sentence level in order to utilize translation memory to reuse the many repetitious fragments that occur in patents. One sentence in a language A may be two or three in B, and what is more, sentences do not always occur in the same order in both the source and the target language because of theme/rheme variation between languages. As a result of these differences, LISA has developed the Segmentation Rules eXchange (SRX) standard designed to ensure the compatibility of segmentation criteria.

5.3 XML Localization Interchange File Format (XLIFF)

XLIFF is the XML Localization Interchange File Format from OASIS, a group of software providers, localization service providers, and localization tools providers including Oracle, Novell, IBM/Lotus, Sun MicroSystems, Alchemy Software, Berlitz, Moravia-IT, and the RWS Group. The OASIS group describes itself as "having a technical agenda, using a lightweight, open process expressly designed to promote industry consensus and unite disparate efforts" (Cover 2001). The XLIFF standard is intended to give any software provider a single interchange file format that can be understood by any localization provider. "XLIFF is a format to store extracted text and carry the data from one step to another in the localization process" (OpenTag 2004), which affords the potentially multiple localization providers working on individual projects with detailed instructions on procedures and project history throughout the project management chain.

Loosely based on the earlier OpenTag specification, XLIFF exploits the extensibility capacity of XML by borrowing from TMX and integrating XHMTL inside XLIFF documents. By the same token, XLIFF documents can be embedded inside other documents. Developers of other standards (e.g., GILT Metrics, planned W3C internationalization directives such as ITS) are cognizant of XLIFF and are also providing for the integration of multiple namespaces within such XML-based models. Like all XML solutions, it is Unicode-compliant supporting both UTF-8 and UTF-16. It is intended primarily for storing text extracted from software-type files and tagged documents in bilingual format for purposes of documenting and retaining information involving pre-translation, revision history, change tracking, encapsulation and placeholder codes in order to protect inline markup during the localization process. The principal motivating factor behind the creation of the standard has been to create a vehicle for transmitting a wide range of detailed information through the localization process regardless of the variety of vendors and collaborators who may work with the materials involved in a given internationalization/localization project.

Like TMX, XLIFF is an open standard. It is not necessarily a competitive tool, however, to the extent that TMX has emerged as a defining feature of certain software packages. The major reason for this is that XLIFF is a global solution for localization firms rather than being an internal feature provided as a function of off-the-shelf commercial software programs.

5.4 TermBase eXchange (TBX)

TBX is an open XML-based standard format for terminological data recommended by the LISA OSCAR group. It owes its development to a long history, stretching from the original SGML-based TEI effort in the early 90s through its incarnation as MARTIF (ISO 12200:1999) through its reincarnation as the XLT format and finally its adoption by OSCAR as its TermBase eXchange format. Designed to provide support for the importation and exportation of terminological data to and from various terminology management systems, the standard is necessarily complex in order to facilitate the wide variety of formats, data elements, and varying data models that are used by terminological resources. The standard is compatible with ISO 12620 for data categories and ISO 16642, which provides a metamodel for terminology interchange formats.

Uptake of this standard has been slow in comparison to TMX and XLIFF, for several reasons. Perhaps because of its relative complexity, but also because industry has not necessarily been clamoring for a terminology solution, vendors of terminology management systems (TMS) software have not been rushing to include TBX output as a standard functionality of their off-the-shelf software solutions, although proprietary XML-based outputs are becoming more common. Creation of a user-friendly TBX-based editor might possibly foster acceptance of the standard, especially any editor that would allow straightforward conversion from the various XML-based outputs to TBX. High demand for TMX-related solutions, however, is likely to dampen interest in implementing the more complex TBX environment until such time as TMX has been fully implemented. Current discussions in the OSCAR/OASIS environment about embedding terminological references in XLIFF documents (frequently called "TBX-Link") may spur interest in implementing the standard, as may the use of TBX as the output format for future developments in the Wiktionary project.

5.5 Open Lexicon Interchange Format (OLIF)

OLIF, the Open Lexicon Interchange Format, was designed as a user-friendly vehicle for exchanging basic terminological and lexical data, particularly in venues involving NLP systems such as machine translation. Its "user-friendliness" is both an advantage and a disadvantage in that its simplicity precludes the preservation, on the one hand, of more complex embedded structures that are typical of more sophisticated terminology management systems, or on the other, of the variety and complexity exhibited by processable lexical resources. The basic premise of OLIF is critical in environments where human translation plays a significant role along side machine translation because its functionalities are designed to ensure consistent usage and documentation between terminological resources intended for human translators and automatic MT lexicons. Although the standard is widely used in SAP in particular, and it at one time enjoyed something of a relationship with the LISA OSCAR group, it has never really been integrated into the extensible, interoperable environment that is growing up around XLIFF, TMX, and TBX. A new version (2.1) of OLIF is available for download from the Web. It claims to provide features that support "lexical/terminological data exchange, lexicon and terminology management, term extraction, controlled language, information retrieval, and glossary development" (OLIF 2005).

5.6 Lexical Markup Framework (LMF)

The Lexical Markup Framework (LMF) standard, which is being elaborated in ISO TC 37/SC 4, is not strictly speaking designed for localization environments, but it is highly likely to prove useful in a wide range of application areas in the future, including localization.

> [It] describes a high level model for representing data in lexical resources used with multilingual computer applications. The scope of these lexical resources includes both human oriented translation tools, such as machine-readable lexica, and automated human language technologies (NLP), such as machine translation, information extraction, information retrieval, summarization [sic], sentence generation, word clustering, multiword recognition and extraction, word sense disambiguation, proper noun recognition, parsing, and coreference resolution (LMF Draft of 18 September 2004).

LMF is analogous to ISO's 16642-based Terminology Markup Framework in that it presents a "high level model" rather than providing a lexical markup language *per se*. Obviously, the intent of the standard is to provide for optimum interoperability among a wide variety of working environments. The LMF core model will be extremely simple, based on a single word and its associated meanings. This core will then be extensible in the direction of any one of the application areas cited above. One of the goals of the standard is to provide a word-based exchange environment that can serve as the logical contact point for language-related data coming from concept-based terminological resources. Under the SALT project, there had been the intention of providing a link between concept (term) and word-based resources using OLIF as the link, but this notion never really reached fruition. Developers are hoping that LMF will provide the tools for achieving this goal.

5.7 Standards for knowledge discovery, retrieval, and management

William of Baskerville, Umberto Eco's protagonist in *The Name of the Rose* endures villainy, deceit, and the occasional murder in his search for one dangerous and intriguing book, a book that serves both as the repository of forbidden knowledge and at the same time as a unique bibliocidal murder weapon. This is a world where librarians jealously guard the keys to knowledge. Many a reader has no doubt thought amidst the seemingly endless search and sometimes bloody obstacles blocking the path to knowledge that the quest would be significantly simplified by a Library of Congress (LOC) or Universal Decimal Code (UDC) reference and a complete and honest floor plan to the library — although these rational tools would have ruined the mystery. But even in today's well-documented collections, it can be difficult to find a resource if it has not occurred to a cataloguer to classify objects according to the principle that is meaningful for a given search. For instance, a colleague querying the LOC catalog for memoirs as a *genre* found that the only way to find autobiographies was to search by individual authors. Everyone has had the experience of knowing that a piece of information was hidden in a book, but the index, even a good index, refused to cough up that bit of knowledge because the indexer, author, or editor had a different perspective on what was important.

Library catalogs are designed to find concrete resources in physical repositories, and back-of-the-book indexes reference knowledge identified as residing inside resources. Users of digital resources have similar needs, either in the well-tended gardens of digital libraries or the jungle wilds of the open Web. Localizers and curators of the multilingual corpora they generate contribute to enterprise-wide knowledge bases that wise organizations will husband and constantly mine for information, not only to support the translation and localization process itself, but also to maintain and enrich enterprise know-how (see Shreve 2004 for a detailed discussion of information mining techniques with respect to bitext corpora). Figure 2 reflects an effort to create a conceptual network reflecting the many different kinds of knowledge organization systems currently in use for storing, maintaining, and retrieving information in digital environments, as well as automatically mining to extract latent knowledge (Wright 2005).

The Library of Congress MARC (MAchine-Readable Cataloging record) standard (ISO 2709:1996), together with ISO 23950:1998 (NISO Z39.50) facilitates the storage and retrieval of information about both real objects in physical systems or digital resources in library environments. The Dublin Core Metadata set provides standardized data element names used to "supplement existing methods for searching and indexing Web-based metadata, regardless of whether the corresponding resource is an electronic document or a "real" physical object" (DCMI 2005). These standards provide major support for *resource discovery*.

Knowledge discovery inside texts, particularly in digital environments and on the Web, is another issue. Given the dynamic and boundless nature of the Web, creating static indexes such as are found in hardcopy books is impossible. Everyone who uses the Web is familiar with the high recall, low precision results of the all-word indexing employed by too many of the search engines, reminiscent of the 1897 Sears, Roebuck, and Company Consumer's Guide: "If you don't find what you are looking for in the index, look very carefully through the entire catalogue" (cited by Osgood 2005). In order to counter the vagaries of full-text indexing, which is fraught by problems of polysemy and synonymy, various schemes and systems have evolved with the goal of establishing machine-parsable semantic associations between subjects, topics, or terms in controlled vocabularies and terms that occur in texts in corpora and on the Web.

The fundamental standard governing terminology and relations between terms, along with the creation of well-formed definitions, is ISO 704:2000, *Terminology work — Principles and methods.* The theory and practice specified in this standard has affected the development of thesaurus management, metadata registries (of which the Dublin Core is a prime example), and ontologies. Thesauri in the sense used here are controlled vocabularies used for the retrieval of information, generally from repositories that have been appropriately encoded with this same vocabulary. Monolingual controlled vocabularies are governed by NISO Z39-19.2005 and British standard *BS 8723: Structured Vocabularies for Information Retrieval — Guide*, which is undergoing parallel development. Among other features, the new standards address digital issues not contained in earlier standards. ISO 2788:1986 (monolingual thesauri) and ISO 5964:1985 (multilingual thesauri) are still valid, but lack this kind of updated approach. IFLA, the International Federation of Library Associations and Institutions, is in the process of creating a new set of *Guidelines for Multilingual Thesauri*, which addresses current issues in knowledge management.

Terminological concept systems image conceptual relationships between concepts associated with special language vocabulary as it occurs *in situ* in real texts. This is *uncontrolled vocabulary*, even in cases involving standardized terms, which is not news to most translators and localizers. *Controlled vocabularies* are used as keywords and as information retrieval elements, with synonyms and frequently even subordinate or related terms being referenced to a selected search term. *Ontologies* display the same kinds of conceptual relationships found in terminologies and thesauri, with the added capability of providing axioms and rules that can be anchored at nodes or along links (edges) in conceptual hierarchies for the purpose of facilitating automatic inferencing and problem solving on the part of search engines and knowledge processing programs. The W3C is responsible for the OWL Web Ontology Language and the SKOS Simple Knowledge Organization System recommendations, both designed primarily for the creation of semantic knowledge representation systems on the Web. (Anglophones fervently defend *OWL* as the acronym in honor of the dyslexic owl in *Winnie the Poo*, much to the consternation of some humorless non-English native speakers who favor WOL and fail to comprehend the significance of the OWL as a repository of knowledge and wisdom.)

Although efforts are underway to facilitate the automatic generation of ontologies and other KOS, terminological concept systems, thesauri, and ontologies are for the most part top-down systems that preexist in knowledge management environments, which means that knowledge in documents is then linked to these systems via various string and semantic parsing techniques. So-called *high-level ontologies* and highly articulated, multilingual computational lexicons have evolved for purposes of providing universal classification capability for all types of knowledge resources. In brief, Princeton-based *WordNet,* with counterpart systems in nearly forty different language communities, provides something like a universal conceptual hierarchy (WordNet 2005). It goes beyond the scope of this article to enumerate and discuss the various high-level ontology systems currently under development. A major concern is that the organization of knowledge is always facet- or perspective-governed, and different cultural and language communities tend to view the world differently in many diverse ways, which makes it very difficult to generate a truly universal system that satisfies all information an knowledge management needs.

The top-down approach has both advantages and disadvantages. The creation of ontologies in particular and of more complex (faceted) thesauri is highly time-consuming, a factor that is sometimes called the "ontology bottleneck." Furthermore, there remains the problem of hidden or latent semantic knowledge that may be of interest to certain users either because of their unique point of view

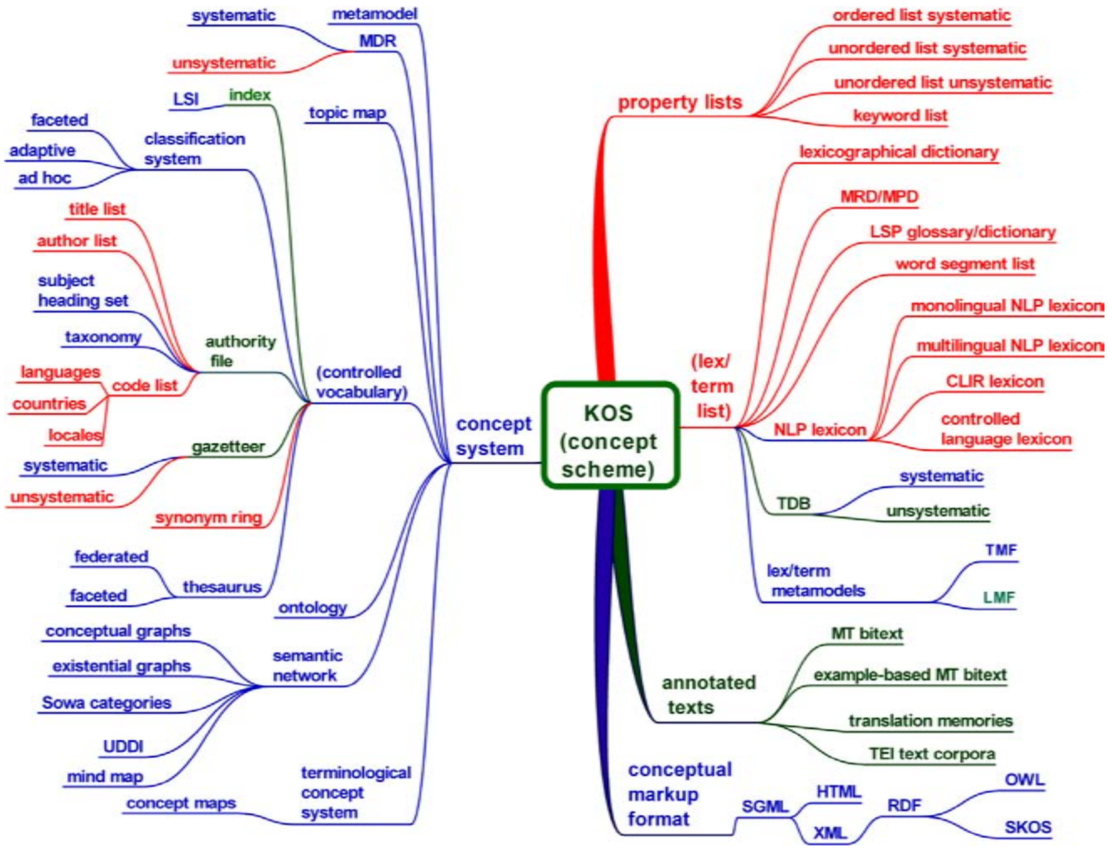

Figure 2. Knowledge organization sources and systems classified according to conventional views (KOS = knowledge organization source/system; MDR = metadata registry; MRD = machine-readable dictionary; MPD = machine-processable dictionary).

(e.g., the memoir as genre example cited above) or perhaps more significantly, because of unforeseen lexical (and hence anticipated semantic) co-occurrences (called as a bit of a misnomer *collocations* by some authors, see the otherwise excellent Passin 2004, 62). Some approaches, such as *latent semantic indexing* (Yu et al., 2002), *adaptive classification* (Calzolari 2002), and *dynamic taxonomies* have provided powerful solutions for knowledge management in industry and not insignificantly with respect to the effectiveness of Web search engines such as Google™. One important strategy that has established itself both in the standards world and in commercial and research venues is *Topic Maps* (Garshol 2003 and 2004; Passin 2004: 60–88), which are standardized in ISO/IEC 13250:1999. Topic maps provide a complex and highly useful approach to establishing semantic relations in external documents for purposes of semantic indexing of resources in digital environments. Passin's introduction provides a clear explanation of how they work, provided the reader has some understanding of XML and RDF representation schemes.

7. Perspectives for the future

As anyone who has actually read the preceding pages must realize at this point, tracking all the standards activities that impinge on the language industry is not a trivial pursuit. Hiring Sisyphus as a research assistant is a serious thought for anyone trying to keep abreast of developments. A recent query to standardizers active in the W3C / Unicode / IETF community asking how decisions are made concerning the placement of standards activities brought mixed and somewhat enigmatic responses. Consensus seems to be that a given project arises wherever a group of experts comes together, recognizes a particular need, and determines (at least in their own view) that they have the collective or in some cases individual expertise to do the job. This state of affairs contributes to the flexibility and power of many of the standards initiatives, but it has its drawbacks in terms of sometimes conflicting or even dueling standards. Various formal and ad hoc liaisons exist between different formal standards institutes on the one hand and less formal industry-oriented programs on the other. For instance, ISO TC 37, Unicode, and the Metadata Open Forum maintain close ties and conduct mutual conferences and workshops, bringing together experts from different communities who share common interests. Establishing some sort of authoritative standards clearinghouse might be one solution, but this notion is not necessarily desirable because such an approach could limit flexibility and no doubt stand in the way of targeting the *right* experts to address new standards problems in a timely fashion. In my view, the best approach would be to expand efforts to maintain a Web site documenting the many standards activities and establish associative links with those that have common interests.

Notes

1. Dabei muss grundsätzlich zwischen Prozessqualität und Produktqualität unterschieden werden. Prozessqualität bezeichnet in der Übersetzungsindustrie die Qualität der Prozesse und Abläufe von der Erteilung oder Annahme eines Übersetzungsauftrags bis hin zur Lieferung des gewünschten Produktes. Produktqualität hingegen bezeichnet die Qualität des Produktes 'Übersetzung'. Sie kann weiterhin in formale Produktqualität und sprachliche Produktqualität unterteilt werden.

2. *GILT* is the LISA acronym for *Globalization, Internationalization, Localization,* and *Translation*, comprising the four constituent areas associated with global information management (LISA 2005).

References

ANSI. 2005a. "ANSI — An historical overview." http://www.ansi.org/about_ansi/introduction/history.aspx?menuid=1

———. 2005b. "Introduction." http://www.ansi.org/about_ansi/introduction/introduction.aspx?menuid=1

Baer, B. and Koby, G.S. 2003. Adapting the ATA framework for standard error marking for translator training. Paper read at the 44th Annual Conference of the American Translators Association, Phoenix, AZ.

Cadieux, P. 2004. "Globalization content model standards." http://www.i18n.ca/publications/GlobalizingContentModelStandards.jpg

Calzolari, N. 2002. "Language resources and semantic Web." The 19th International Conference on Computational Linguistics, Taipei, August, 2002. http://utrecht.elsnet.org/roadmap/docs/coling2002-nc.ppt#1

Carless, S. 2004. "Lost in translation: Japanese and American gaming's culture clash." *Gamasutra*, January 21. http://www.gamasutra.com/features/20040121/carless_01.shtml

Colina, S. 2003. *Translation Teaching, From Research to the Classroom: A Handbook for Teachers*. Boston: McGraw-Hill.

Constable, P. and Simons, G. 2000. "Language identification and IT: Addressing problems of linguistic diversity on a global scale." *SIL Electronic Working Papers 2000-001*. Dallas, TX: SIL International. http://www.sil.org/silewp/2000/001/SILEWP2000-001.html

Cover, R. *Cover Pages. Hosted by OASIS*. http://xml.coverpages.org/

———. 2002. "Localization Interchange File Format TC." http://xml.coverpages.org/XLIFF-TC-20020122.html

———. 2003. "XML Localization Interchange File Format (XLIFF)." *Cover Pages: Technology Reports*. http://xml.coverpages.org/xliff.html

———. 2004. "Language identifiers in the markup context." *Cover Pages: Technology Reports.* http://xml.coverpages.org/languageIdentifiers.html

———. 2005. http://xml.coverpages.org/languageIdentifiers.html

Crystal, D. 2001. *A Dictionary of Language.* 2nd ed. Chicago: University of Chicago Press.

Davis, M. 2004. "Unicode Technical Standard #35 (UTS#35): Locale data markup language (LDML)." http://www.unicode.org/reports/tr35/

DCMI. 2005. "Dublin Core Metadata Initiative FAQ." http://dublincore.org/resources/faq/#whatisaresource

Delisle, J., Lee-Jahnke, H. and Cormier, M.C. (eds). 1999. *Terminologie de la Traduction*. Amsterdam/Philadelphia: John Benjamins.

Doyle, M. 2003. "Translation pedagogy and assessment: Adopting the ATA's framework for standard error marking." *ATA Chronicle* 32 (11): 21–28.

Dunne, K. 2004. "Putting the 'Professional' in Professional Translation." *ATA Chronicle* 33 (11): 15–19.

Felber, H. 1984. *Terminology Manual*. Paris: Unesco/Infoterm.

Garshol, L.M. 2003. "Living with topic maps and RDF: Topic maps, RDF, DAML, OIL, OWL, TMCL." Ontopia. http://www.ontopia.net/topicmaps/materials/tmrdf.html

———. 2004. "Metadata? Thesauri? Taxonomies? Topic Maps! Making sense of it all." Ontopia. http://www.ontopia.net/topicmaps/materials/tm-vs-thesauri.html

Goldfarb, C. 1996. "The roots of SGML — A personal recollection." http://www.sgmlsource.com/history/roots.htm

House, J. 1997. *Translation Quality Assessment: A Model Revisited*. Tübingen: Gunter Narr Verlag.

ICU. 2004. *Locale Explorer*. http://oss.software.ibm.com/cgi-bin/icu/lx

Juran, J.M. and Gryna, F.M. 1988. *Juran's Quality Control Handbook*. 4th ed. New York: McGraw-Hill.

LISA. 2004. "Industry standards." http://www.lisa.org/standards/

———. 2005. "Frequently asked questions." http://www.lisa.org/info/faqs.html#gil

Kemmer, S. 2003. "Chronology of events in the history of English." http://www.ruf.rice.edu/~kemmer/Words/chron.html

Klaudy, K. 1995. "Quality assessment in school vs. professional translation." In *Teaching Translation and Interpreting 3: New Horizons. Papers from the Third International Language Conference, Elsinore, Denmark, 1995*, C. Dollerup and V. Appel (eds), 197–207. Amsterdam and Philadelphia: John Benjamins.

Martin, B. 2001. "Is your content globally correct? Best practices for enterprise, multilingual content generation, management and delivery." *AIIM 2001 Conference Presentations*. Silver Spring, MD: AIIM International. http://198.77.178.130/events/aiim2001/presentations/100Martin.pdf

Melby, A., Manning, A.D. and Klemetz, L. Forthcoming. "Quality in translation: A lesson for the study of meaning."

Neubert, A. and Shreve, G.M. 1992. *Translation as Text*. Kent, OH: Kent State University Press.

Nord, C. 1997. *Translating as a Purposeful Activity. Functionalist Approaches Explained*. Vol. 1 of *Translation Theories Explained*. Manchester, UK: St. Jerome Publishing.

OLIF. 2005. "The versatile XML language data standard." http://www.olif.net/

OpenTag. 2004. "Text Filtering > XLIFF." http://www.opentag.com/xliff.htm

Osgood, M. 2005. "Back words indexing: Back-of-the-book indexing for publishers and authors." http://backwordsindexing.com/Intro.html

Passin, T.B. 2004. *Explorer's Guide to the Semantic Web*. Greenwich, CT: Manning.

Schnitzlein, M. 2003. *Zum Aussagewert von Qualitätsnormen und Qualitätssicherungssystemen für die Translationsqualität — eine exemplarische Analyse*. Diplomarbeit, Applied Linguistics and Translation/Interpreting, University of the Saarland, Saarbrücken.

SDL International. 2004. "TMX: Maximizing the return on your translation memory investments." *The LISA Newsletter*, July 2004. http://www.lisa.org/archive_domain/newsletters/2004/2.3/sdlAdvertorial.html

Shreve, G. 2002. "Knowing translation: Cognitive and experiential aspects of translation expertise from the perspective of expertise studies." In *Translation Studies: Perspectives on an Emerging Discipline*, A. Riccardi (ed), 150–171. Cambridge, UK: Cambridge University Press.

———. 2004. "Corpus enhancement as an internationalization strategy for large-scale intercultural communication." *Logos and Language* 4 (2): 31–48.

SIL International. 2005. "Introduction to the printed volume." http://www.ethnologue.com/ethno_docs/introduction.asp

Techstreet. "The what, why, and how of standards." http://www.techstreet.com/whystandards.tmpl. Originally published as Breitenberg, M.A. 1987. *NBSIR 87-3576. The ABC's of Standards-Related Activities in the United States.* Gaithersburg, MD: National Institute of Standards and Technology. http://ts.nist.gov/ts/htdocs/210/ncsci/stdpmr.htm

Unicode. 2005. "The Unicode standard: A technical introduction." http://www.unicode.org/standard/principles.html

———. 2005. "Common locale data repository (CLDR) project." http://www.unicode.org/cldr/

Vinay, J.-P. and Darbelnet, J. 1995. *Comparative Stylistics of French and English: A Methodology for Translation.* Amsterdam/Philadelphia: John Benjamins.

Yu, C., Cuadrado, J. and Ceglowski, M. 2002. "Patterns in unstructured data discovery, aggregation, and visualization: A presentation to the Andrew W. Mellon Foundation by J. Scott Payne." National Institute for Technology and Liberal Education (NITLE). http://javelina.cet.middlebury.edu/lsa/out/cover_page.htm

WordNet. 2005. "WordNet 2.0 search." http://wordnet.princeton.edu/cgi-bin/webwn

———. 2005. "WordNet: A lexical database for the English language." http://wordnet.princeton.edu/cgi-bin/faqview.cgi

Wright, S.E. 1991. The role of terminology in translation-oriented quality assurance. Paper read at the 32nd Annual Conference of the American Translators Association, Medford, NJ.

———. 1998. "Trends in language engineering." *TAMA '98, Proceedings of the 4th TermNet Symposium "Terminology in Advanced Microcomputer Applications, Tools for Multilingual Communication*," 3–23. Vienna: TermNet.

———. 2001. "Convergence Technologies." *TAMA '01: Terminology in Advanced Microcomputer Applications, Fifth TermNet Symposium*. Invited keynote address presented at Antwerp, Belgium. http://www.nlterm.org/

———. 2005. "Standards for Content Creation and Globalization." http://appling.kent.edu/ResourcePages/LTStandards/Chart/LanguageTechnologyStandards.htm

———. Forthcoming. "Coping with indeterminacy: Knowledge organization systems in digital environments." In *Festschrift for Heribert Picht*, Bassey Antia (ed). Amsterdam and Philadelphia: John Benjamins.

Wright, S.E. and McClure, S. 2003. "The state of language industry standards and their impact on information technology for the global enterprise." In *IDC Data Watch*, February, Document number 28752. http://www.idc.com/getdoc.jhtml?containerId=28752 (no longer available)

PART 7

Rethinking the paradigm

Melding paradigms

Meeting the needs of international customers through localization and user-centered design

Susan M. Dray and David A. Siegel

Introduction

In the global economy, products are designed in a given country to be sold and used in countries around the world. When we design products, services, and Web sites, especially those for international users, it is important to start with a deep understanding of *all* users — both domestic and international. This includes understanding how they are similar — and different — in different parts of the world, and then using that knowledge to design a product or products that will "work" for each of them. Clearly, this approach goes beyond simply localizing products. It touches the very heart of what a product is intended to be used for, and how that product fits into users' lives regardless of where they live. And it requires that the design be truly "international." This is an enormous challenge.

Because "design" can and does mean different things to different people, it is necessary to clarify exactly what we mean by the term. When we speak of "design," we are not merely referring to a product's appearance but rather to the entire process by which products are conceived and developed, from the earliest stages of planning all the way through to launch and subsequent support. Our holistic definition focuses on the approach and the process, and not merely on the output of that process.

Unfortunately, most companies do not adopt a holistic approach when creating their products, services and Web sites. Instead, products are typically developed to fit into a gap in the market, whether or not that gap corresponds to a true user need. For example, if a company notices that its rivals have launched a new feature or a product, they may add something similar to their product line simply to keep up with the competition. Alternatively, they may focus on technology as an end unto itself and see all kinds of new, exciting or "cool" things they can do with the technology, forgetting that people might not need, want, or be able to use the product that results. They may design something for the company's own convenience, such as an online sales tool or a technical support knowledge base, in the hope that this will help the company save money or other resources — without knowing whether these "solutions" will in fact meet the actual needs of customers, especially those in other countries. Such design decisions are all driven by producer concerns, rather than by user concerns. And all too often, the responsibility of adapting a product for international users is abdicated, and the hard work of doing this is left to localization professionals, who often first see the product very late in the process, when it is too late to assure success in the market because there is only time for minor changes.

Fortunately, there is an alternative to the dysfunctional producer-centered design process, namely user-centered design (UCD). After detailing what is lacking in the current design process, we will discuss what UCD is and show why it is important (and how it can even help save money). Finally, we will provide some examples of the methods used in UCD, illustrated by representative UCD projects in which we have participated.

The problem with conventional product planning and design practice

Because producer concerns and motivations predominate in conventional product planning, the design of products tends to reflect the mindset of the producers, rather than that of users. Indeed, several factors have disproportionate weight in product development and design despite having little or nothing to do with true user needs, behaviors, or circumstances.

Technocentric design

Much of the motivation guiding design and development of technology today is "technocentric." In other words, the product planning process begins with the team identifying a novel or "cool" technological capability. Once the product is built, the company then tries to persuade users to adopt it. In other words, companies often build new products or add new features to existing ones because they *can*. There are many examples of this in products today. For instance, all the formatting options and editing features of word processing software are rarely used or understood (even if the user looks them up in the Help). Many of these are there because of this technocentric process. The Web is also full of examples. Consider the "flashy" Web site that has sophisticated graphics and animations, but which takes longer for users to download, especially users on dial-up connections.

Subtler forms of technocentrism are also very common. Often, design of the user interaction inappropriately assumes technical knowledge on the part of users, or requires them to focus on managing the technology rather than on achieving their goals. A classic example of this is the very common situation in which navigation in an interface requires an understanding of the underlying architecture of the software or other technical factors. For instance, we recently worked on a specialized printer. It originally had a 5-step calibration process which required the user to understand technical details of how this device used a number of variables to calibrate printing alignment. In addition, there were three different calibration modes which differed from each other in complex ways. Even technically-oriented members of the team who were not the calibration specialists were confused by how the variables affected each other. Not surprisingly, the calibration interface was designed by the calibration specialists who did not realize that this complexity might be an issue for users, let alone that there was another way to design the interface. We redesigned the process into two steps that merely require the user to identify two characteristics of the document to be printed (i.e., things with which the users were very familiar).

Similar examples are common on the Web. For instance, many sites that offer audio file downloads require users to select the format used by their player, and if the users guess wrong, an error message displays telling them of this incompatibility. This problem can affect users who are new to the process in a way that stops them from exploring further. Even sophisticated users often infer from the error message that they simply can't download that particular file.

Feature creep

Conventional design is also driven by companies' desire to keep up with or differentiate themselves from the competition. This leads to framing so-called "value propositions" (the unique value to consumers from which revenue will be derived) in terms of the number of different "cool" things the product offers. Also, in many companies, teams are rewarded based on whether or not their feature is included in the product. Such practices in turn drive teams to push for including the feature they designed in the final product, whether it causes problems for the users or not.

Producer convenience

Design is too often driven by concerns that have to do with making the producer's life easier. These may include focusing on the convenience or efficiency of the code writing process for software, ease of providing support or administration, reusing existing code, or leveraging an existing platform or technology where there is company expertise or experience. One common manifestation of this phenomenon in the software industry is the attempt to solve a design problem by adding additional user-selectable options, as opposed to providing more fundamental solutions. This is one way that designers sometimes try to sidestep thorny design decisions. Unfortunately, these options themselves add another layer of complexity and abstraction that makes it harder for "normal" users to understand and use the product. In his classic book, *The Inmates are Running the Asylum*, Alan Cooper (1999) provides excellent insights into the ways in which programmer convenience and incentives can dominate the design process.

Unanalyzed collections of user requests and complaints

Although most companies collect user complaints, they do not always analyze them to understand the basis of the problem(s) that led to the complaints in the first place. Complaints are definitely clues that there are problems in a product, but they do not convey the context of the problems users are experiencing, nor do they indicate what to do to fix the problems.

User requests have similar pitfalls. People make requests based on what they need to do today, or what they would like to see in the future, but users are not designers. User requests do not generally reveal to designers what is driving the request, nor do they tell the team whether or how to implement a solution or a feature to address the request. User requests also cannot tell the designers how this element or feature will impact the other parts of the product, making it difficult for them to provide an integrated solution. There is a dark side to user requests: they often take on a life of their own, especially when they come from important users. Many companies have "user advisory boards" of influential users or stakeholders, and requests from this type of group often have an untoward impact on product planning. User requests and complaints should be fodder for the design process, but not directives to the design team.

Such factors will probably always have some influence on product planning and design. But when they become the primary focus, the consequences for product design can be very negative, leading to products and technology that are flashy and filled with features that few people can or do actually use. In extreme cases, the products may actually be unusable, or at least not useful. Additionally, these products may not "fit" into people's lives. Ultimately, therefore, these products are likely to be eclipsed by products that may have similar functionality, but which fit better with users' tasks, goals, and passions.

What is user-centered design?

User-centered design (UCD) begins from a very different premise, namely that if companies are to design products that will truly meet users' needs, they have to start by gaining deep understanding of who their users are, and, perhaps most importantly, how the new product or service will fit into the cultural, social, technical, and physical contexts of the intended users' lives. UCD is a product planning and design process that uses specific methodologies to introduce user considerations into all stages of the design process. It does this in ways that help balance technology and producer considerations with user and customer considerations. Unfortunately, even if the product team wants to be sensitive to user needs, it is almost impossible for them to understand through intuition alone how users will respond to a product or how it will fit into the context(s) of their lives. UCD is not just a change of attitude. Designing in a user-centered way requires a fundamental shift in the very process of design, towards users, beginning with user research and prioritizing user research throughout the design process.

Why other approaches to obtaining user input fall short

It might seem that the most straightforward way to obtain user input is simply to ask people what they like or dislike. Indeed, many marketing departments do collect this kind of information through surveys and focus groups. While this may seem logical on the face of it, when we look deeper at the type of data that these methods yield, it becomes clear that although such data may be useful for identifying information to guide branding or messaging, they do not provide the type of information that is needed to achieve better design. Both surveys and focus groups rely on self-report. In each of these methods, people are asked questions to which they provide answers. Unfortunately, self-report has significant limitations, especially in providing information that will be useful in design. For one thing, there are psychological limits to recall and introspection: people tend to recall things that are more recent, or more salient, and forget other things that may be critical in helping designers achieve better designs. In a recent experiment, we asked people to describe to us how they brushed their teeth, and then we filmed them doing just that. Interestingly, the video captured many things that people had failed to mention in their verbal descriptions of how they brushed. People also have a tendency to give "reasonable" replies that are often only partial answers. In addition, people typically overestimate their motivation to change their behavior, and it is difficult for them to imagine accurately what these changes might entail in their lives. For these reasons, it is not enough just to ask people what they think or feel. We must instead observe what people actually *do*, and use these observations to augment what they say.[1]

Why is UCD important?

The shift to UCD can be challenging for a design team, and without a strong reason to change, designers are not likely to adopt this new approach. Companies therefore need to understand the benefits of UCD.

UCD matters because design is most successful when it is based on what real people need and want. Products designed using UCD methods reflect users in their very conception. Users' tasks are supported better, and therefore, these products "fit" users' lives more naturally and the design of the product makes interacting with them more intuitive. Products designed using UCD methods are more likely to be useful, usable, and desirable.

Although it can be difficult to quantify the value of UCD because its wide-ranging impacts are difficult to measure, many have made the case for its return on investment (ROI). Perhaps the most commonly cited "figure" for the ROI of UCD is IBM's "rule of tens" (IBM 2004a) which states that a problem that costs \$1.00 to fix during design will cost ten times more (or \$10) to fix during development, and another ten times that cost (or \$100) to fix after the product's release. Other cost–benefit analyses focus on reduced support and training costs, improved user efficiency and accuracy, increased likelihood of completing tasks such as purchasing from retail Web sites, increased discoverability of product functionalities and their benefits resulting in more customer attachment and loyalty, and so forth (see Bias and Mayhew 1994, and Bias and Mayhew 2005 for many examples).

UCD for international markets

The cost–benefit arguments in favor of adopting UCD early in the design process are even more compelling when designing for international markets. First, in the international context, the risks of making fundamental errors at the level of product concept and value proposition, conceptual design, and major functionality are greatly increased in the absence of early UCD research. Clearly, we can't assume that "one size fits all" and that products created for one market will work in another. When developing products for people in other countries, designers must be even more careful about the assumptions they make. Second, if a mistake requiring changes to an international product is made, the cost of reworking it may be substantially higher than for a domestic product, especially if the mistake is not noticed until late in the process, because of the wider variety of additional downstream costs associated with international roll-outs (such as localization costs).

Localization — or fitting a product to the users and context of another country — is not simply a matter of translation, adapting the interface to fit local information display conventions, or visual design preferences that are different from those we are used to. We also need to understand how people work and live in other places, so that the localized product will fit into their lives. When we learn about this, we may decide that the very product concept has to change for localization to even be possible.

For instance, in Latin America we did a series of studies of a Web site that, among other things, provided information for troubleshooting a computer product, retrieving software updates, and purchasing equipment online. During the usability evaluations, we noticed that users seemed to be struggling with certain tasks. At first, it seemed that the users' difficulties might be due to problems with the layout or translation, but ethnographic visits with people in their homes suggested the true roots of the problem were much deeper. In the local context, the people whom we recruited because they had a computer at home and dial-up Internet access occupied a very different socio-economic niche from that of families who owned similar computer equipment in the U.S. where the company had its headquarters. In the U.S., such people would be considered "mainstream" users, but in the Latin American countries we were studying, they constituted a very privileged group. Not only did they often have servants in their homes, but their expectation was that service technicians would also make house calls to address computer problems. We began to realize that the design implicitly assumed a universal mindset of "self-help" and "do it yourself," which, in reality, might differ significantly from one country to another. Insights about issues such as these led to a redefinition of the business purpose of the site in order to better adapt it to the local cultural context.

Once we know what users' needs and motivations are, and we understand the opportunities and constraints that their contexts provide, we still need to understand how to design products so that people will be able to use them successfully. Here again, UCD plays a key role. Understanding *how* people perform tasks helps us to understand how to design "usable" products that can support users in doing their tasks. "Usable" products are those which are easy to learn; efficient, straightforward and satisfying to use; memorable; and which make it easy to recover from errors that may occur during usage. UCD includes methods to evaluate usability iteratively as the design evolves. We will describe these in more detail later, but first, we will examine the relationship between UCD and localization.

UCD and localization

The relationship between localization and UCD should be synergistic. Clearly, both professions are striving to create products that "work" in international locales. And, just as clearly, both face some common challenges and misunderstandings of the degree to which they can contribute to international design. Both struggle to be included in the process earlier than is typically the case. Just as localization professionals need to obtain material for localization earlier, UCD professionals need to be able to collect user data earlier in the design. Similarly, both professions need to be considered key members of the team for the duration of a project to meet their objectives. Therefore, there is a huge opportunity for UCD and localization professionals to partner and to support each other's bids for enhanced access to the design process. These common challenges are summarized in Table 1.

Table 1. Common challenges facing UCD and localization professionals.

Localization	User-Centered Design (UCD)
"It's not just translation!"	"It's not just screen design!"
Getting material early enough to localize fully	Getting user data at the right point (in the design process)
Understanding product *and* context of use	Same plus technical constraints such as current users' technical experience, etc.
Inclusion on team for cultural and technical input	Influencing design through all its phases

Imperative but inadequate processes

Similarly, a number of practices are necessary — but *not* adequate — to achieve both excellent localization and excellent UCD. Again, they are remarkably parallel. Following guidelines, conventions, and formats, while a potentially important component of a larger strategy or process, does not ensure either good localization or good UCD. Neither does simply including local representatives, user representatives or subject-matter experts (SMEs) on design teams. Finally, while back-translation and usability testing of final designs are good ideas, neither is sufficient to ensure good localization or good usability because they occur far too late in the process to be effective. At this stage, fiscal and/or time constraints generally preclude the possibility of taking corrective action. These parallel necessary-but-inadequate practices are summarized in Table 2.

Table 2. Necessary but inadequate current practices.

Localization	UCD
Follow language and style guides	Follow User Interface (UI) guidelines
Translation and formats	UI conventions
Local representatives	User representatives
Local Subject-Matter Experts (SMEs)	SMEs
Back-translation	Single final usability test

Re-examining the design process

Clearly, both localization and UCD stand to benefit from working together, but to make this happen, we need to understand the barriers to making such collaboration a reality. The problem that we both must address is not the content or value of our own fields, since both contribute greatly to a product's usefulness and usability, but rather the very process by which products are designed and developed.

How UCD conceptualizes the phases of design

As stated earlier, "design," especially user interface (UI) design, is not simply an end-of-product-development afterthought that occurs when a user interface person determines the look and feel of the user interface to be grafted onto the product: design starts — or should start — at the very conception of the product. Unfortunately, most products designed today still suffer from a technocentric bias. As described above, this means that the focus is not on the user — least of all the *international* user — but rather on the technology itself. In addition, the design process itself is usually driven by the designers and developers, who have a culture of their own. Designers see the world differently from ordinary end users: they tend to be more feature-focused, excited about new technology, and tolerant of design complexity than the people for whom they are (ostensibly) designing. In fact, designers usually are unable to recognize the complexity they create and are often ignorant of the downstream repercussions of their design decisions. Therefore, without a conscious "reality check" at all stages in the evolution of the product, the design process naturally tends to yield products that designers and developers think are "cool," but which "normal" users find baffling.[2]

The phases of design in UCD are presented graphically in Figure 1, and also described in Table 3.

Design starts with the very earliest discussions of a future product. During this initial phase, called "Conceptual Design," the design team must determine what their product will be and what it will do, which entails mapping out core

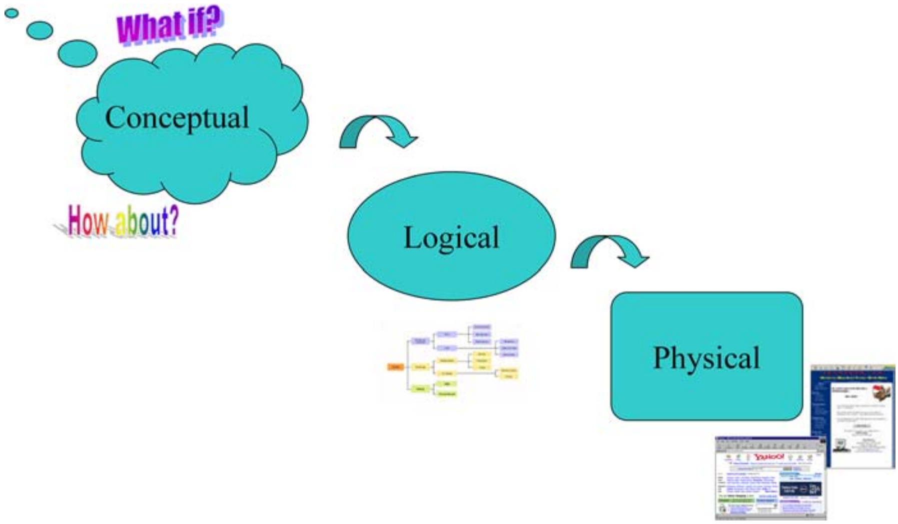

Figure 1. The UCD design process.

Table 3. Key tasks during design by stage.

Design stage	Issues and tasks
Conceptual Design	• Determine the role this product will play in people's lives • Address the fit of the product with the dynamics of current practices and tools • Determine how this product will fit into people's social, cultural and physical contexts • Identify core functionality • Define high-level organizing concepts
Logical Design	• Map out the logical categories of information and groupings of functionalities • Map out logical relationships among these categories and groupings • Map navigation flows and construct interaction design
Physical Design	• Choose appropriate controls (buttons, hyperlinks, etc.) • Map out layout, colors, graphics • Create icons and imagery to support navigation and use

functionality and identifying the problem(s) that the product is being developed to solve. In a UCD process, this phase will also describe how this product will "fit" into people's lives — "people" referring to users in *all* of the product's target markets around the world.

This process requires understanding users thoroughly and deeply. One of the most critical tasks during conceptual design is understanding the users' "mental models." A person's "mental model" is the way that s/he thinks about and

Figure 2. What is this for?

conceptualizes tasks, as well as things in his/her environment. This mental model helps users transfer knowledge they have built up in one domain to another.[3] For instance, if you are shown an item such as that in Figure 2, you probably have some idea of how to use it, because even though you may not have used that individual object before, it looks like other implements you have seen and used before. In fact, you have a name for it — a pen — and the mental model you have of it probably includes the knowledge that it is used for writing. You may not know everything about it, such as whether it writes with blue or with black ink, or whether you click on the end or twist the barrel to extend the pen tip, but you have a sense of the various possible ways in which you might get the pen to "work." If, however, instead of extending the tip when you twist the barrel, this object squirted water at you, it would violate your mental model of a "pen" and it would be either funny or frustrating, depending on whether or not you were relying on this object to write something important. Indeed, violating our mental models is the basis for many "practical jokes" — as well as for much of the "unusable" stuff in our world.

Mental models help make new things more understandable and therefore more "usable." Conversely, when mental models are violated, things are less usable. Indeed, the closer a product's conceptual design is to the mental model that users have of the task it is intending to support, the more "intuitive" the product will be.

Therefore, without understanding users and without learning about their mental models, it is almost impossible to create a conceptual design that will be truly usable. Unfortunately, once the conceptual design is set, it is extremely difficult, and often impossible, to change. If a product's conceptual design does not match the intended users' mental models, this type of mismatch will make a product truly unusable by many — or even most — users. Clearly, therefore, incorporating information about intended users at this early stage of design is critical to the ultimate success of products. In today's markets, success generally connotes profits. However — and paradoxically — if "success" were conceived not merely in terms of unit sales but also in terms of usability, the sales numbers and profits might be even higher.

As the definition of the product advances and development moves ahead, the team moves on to the "Logical Design" phase, as shown in Figure 1. During this phase of development, the relationships between the functions are mapped out and specified, giving the team a first pass at what the potential navigation may be. This, in turn, determines the interaction design — or how a person might move through the product's functionality. This will guide the actual interface design during the next stage of development. Again, it is the users' logic that should be used to structure the logical design. Too often, the logical design of an application mirrors the organizational structure of the teams that have developed different aspects of the design, who may or may not have even talked with each other. As noted above during the discussion of the technocentric specialized printer, teams are often organized according to their technical specialties, and there is often no incentive for them to talk or coordinate their approaches to the structure of their own section of the application. It is not unusual to find different logical structures in different sub-sections of a complex product as a result. This typically results from a lack of an overall logical design as well as from a lack of understanding of the users' logic.

Once a product reaches the "Physical Design" phase, it is typically quite well defined. Because the functionality and navigation have been defined in previous design phases, the designers then tackle the challenge of communicating this functionality and navigation to users through the visual design and layout. At this stage, designers choose the appropriate controls, such as text boxes, buttons, or hyperlinks, and arrange them on the page (if it is a software product or Web site) or on the body of the product (if it is a hardware product).

Unfortunately, if there are significant underlying problems with the conceptual and/or logical designs, it is usually virtually impossible to solve them in the physical design. Creating a UI that has the "correct" controls and a pleasing layout is insufficient to fix underlying functional or navigational problems. It is not unusual for design teams to seek the assistance of UCD professionals after the Physical Design has already been well elaborated. At this point, however, there is

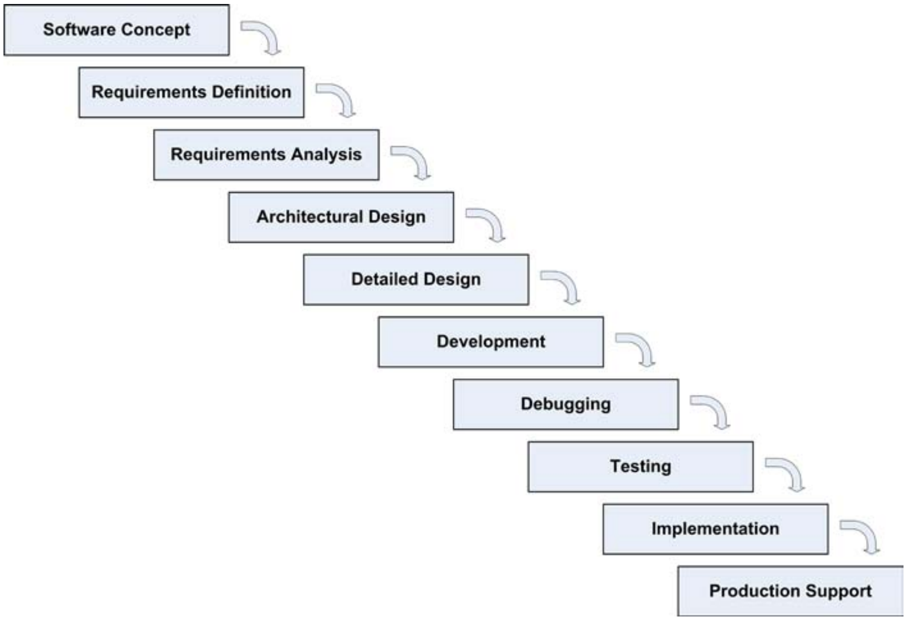

Figure 3. Traditional "waterfall" methodology.

often not much that can be done. As one colleague rather colorfully puts it, "they are bringing me in to put lipstick on the pig."

Conceptual, Logical, and Physical design are abstractions that provide a way of talking about different levels of design. Design and development practice is also organized into sequences or phases of activity. The traditional model for organizing the development of technology products is not well suited to handling the interdependent levels of conceptual, logical and physical design. This approach has been frequently referred to as the "waterfall" method, so named because one phase of development activities "flows" into the next in a linear sequence of interrelated dependencies. Figure 3 shows the Waterfall Methodology in graphical form.

The key weakness of the Waterfall Method is that it does not allow for any significant iteration. In other words, there is no time to make changes to outputs from previous phases. For instance, if during the Detailed Design phase the team finds out that a key requirement (specified several phases earlier) is incorrect, there is no way to go back and rethink it. So, for example, if the team discovers that there is a whole new type of call that customer service reps regularly have to deal with but that was not specified during Requirements Definition, it is too late to address that requirement in the design if the team is using the Waterfall Method. This was a deficiency even in mainframe days, when this process was created. The reason that iteration is so important is that although conceptual design and logical design are more fundamental than physical design, it is very hard to evaluate them except through some physical instantiation. For this reason, it is essential that the design methodology allow for testing of early, low-fidelity prototypes at a point in the process when conceptual and logical design problems can still be rectified with relative ease.

A better way to design

UCD aims to ensure usefulness, fit, and usability by incorporating the appropriate types of user data throughout the design process. This begins in the earliest stage of design, continues during the evolution of the product concept and requirements, and extends throughout the entire process of product development. For international products, all of these phases need to be grounded in *international* user and context information instead of only domestic data.

The International Organization for Standardization (ISO) has formulated an international standard that describes the basic process of UCD. This standard (ISO 13407. *Human-centered design processes for interactive systems*), defines a general process for integrating human-centered activities throughout a development life cycle. In this model, four activities form the main cycle of work and are conducted iteratively:

1. **Specify the users and the context of use**: Identify the people who will use the product, the use(s) to which they will put it, and the conditions under which they will use it.
2. **Specify requirements**: Define any business requirements or user goals that must be met for the product to be successful.

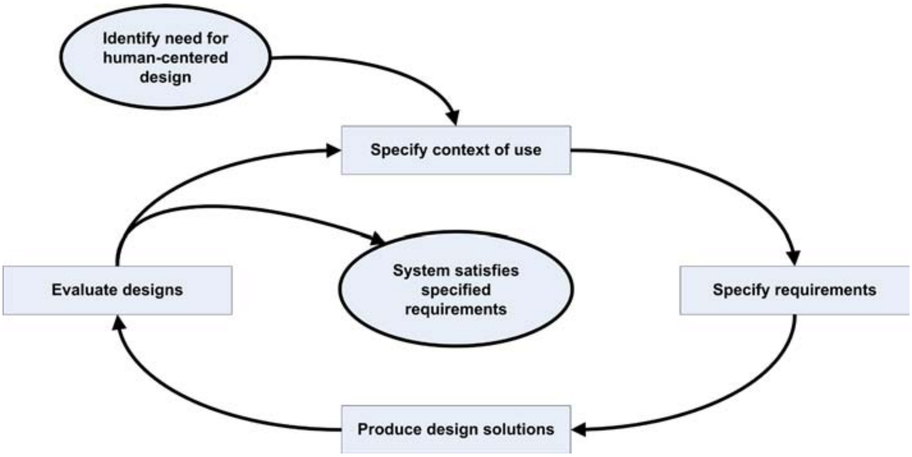

Figure 4. The human-centered design process (ISO 13407).

3. **Produce design solutions**: This part of the process should be done in stages, building from a rough concept to a complete design.
4. **Evaluate designs**: The most important part of this process is evaluation — ideally, usability testing each design solution with actual users. Evaluation is the key to success because it catches problems (or potential problems) early and allows a team to change direction if necessary in order to meet user goals.

A graphical representation of this process is shown in Figure 4.

UCD tools and techniques

Of course, UCD requires not only that users, the context of use and the requirements all be specified, but also that these specifications be based on *data*. UCD includes methodologies for collecting data about users, the usage context, and the performance of the evolving design. In this section, we will review some of the techniques that can help focus the design process on the user and provide examples of some of the ways we have used these techniques to help our clients design more usable products. While we will introduce three of the most useful UCD techniques for products intended for international use, this list is by no means exhaustive.[4]

Ethnographic or field studies

Field studies are a powerful UCD tool. In this type of user study, trained researchers conduct extensive observations of users in their natural environments, while the users are carrying out activities that are directly or indirectly relevant to the domain of interest.

These studies differ from the ethnographies sometimes done by market researchers in that UCD ethnographies focus on observed behavior more than on self-report. In a UCD ethnography, there is certainly verbal interaction between the researcher and participants because interviewing may be necessary to help the researcher interpret what she/he is seeing. However, UCD ethnography goes to additional lengths to make sure that the self-report is grounded in samples of behavior. One technique, known as "contextual inquiry," is a style of interview conducted while the user is carrying out activities in his or her natural environment. "Artifact walkthrough" is a different technique, in which the researcher focuses on some specific output of a user behavior, such as an email, printout, spreadsheet, or document (the "artifact") in order to explore its history and the role it plays in the participant's work or home life. These techniques typically enable the researcher to uncover additional aspects of the usage dynamics that are explored in turn. Typically, a field study will include a flexible and shifting mix of these methods.

Field studies are structured by the use of a "focus structure," which is built up by collecting and grouping all of the different things that a team is trying to learn about, and figuring out the logical relationships between these groups. The resulting structure allows the team to follow the flow of a visit while paying particular attention to the things that are most important to discover. It also allows the facilitator to guide and direct the questioning to delve more deeply into the most important topics or tasks while letting the conversation remain more superficial in other, less critical areas. Therefore, this "focus structure" is like the skeleton for the field study.

Ethnographic studies are most useful in the early stages of product development, when they can be used to help shape product direction. Because they occur in the *user's* environment, they can yield rich clues as to both the user's mental models and the factors in the environment that will shape the user's experience with the ultimate product. For international products, ethnographic studies are an excellent way to find out about users and to provide a team with meaningful feedback and a "reality check" from the users themselves about the product concept before they have invested significant funds in the development of a product that may have issues serious enough to compromise its chances for success.

Ethnographic studies are also particularly critical when a team is designing a product for international use. As a case in point, when Hewlett-Packard was designing their Infiniium Digital Oscilloscope (see Figure 5), the company utilized a wide variety of user-centered design methods. However, when conducting usability evaluations with design engineers in Japan, we discovered some ambivalence toward this new oscilloscope, even though the engineers liked the design and used it easily in the usability lab. A visit to Japanese R&D labs instantly revealed the source of the hesitation. Compared to engineering benches in the U.S. and Europe (where we had also conducted usability tests), Japanese engineering benches are much smaller. The product we were testing, like that shown in Figure 5, used a mouse as a pointing device — yet there was no room on the Japanese engineers' benches for an additional mouse! As a result, Hewlett-Packard delayed its introduction of the Infiniium into the Japanese market by roughly 6 months while they explored design alternatives to the mouse. When the revised model of the Infiniium was introduced, it was embraced by Japanese R&D engineers. The project leader later confided that had the Infiniium been released without an alternative to the mouse in Japan, it would surely have failed (Dray and Rowland 1998).

In another study, we worked with a client who was creating mapping software for use in businesses around the world. We conducted an extensive ethnographic study in several European countries to discover the appropriate way to present geographic information before the team designed the European version of the

Figure 5. Infiniium 54832D Digital Oscilloscope produced by Agilent Technologies (formerly part of Hewlett-Packard).
Photo courtesy of Agilent Technologies, Inc (www.agilent.com).

application. In this case, the team needed to understand how to display the actual geographic information itself, not merely to ensure that the information (such as addresses or units of measure) was formatted correctly, which was clearly a standard localization issue. However, deciding how to "chunk" information about a specific area, determining whether users perceived a given region as being close to or distant from them, and ascertaining how to display various types of geographic information in ways that were understandable was critical for this application, and this type of information clearly depended on specific local circumstances and behavior patterns. It also had different relevance for different purposes. For instance, if the company was using the geographic information to schedule deliveries by truck or lorry, distances were perceived differently than if the information was used to map out quadrants for mail campaigns. Without studying the issue specifically, it was impossible to determine (or even guess) which locations a person in another part of the world would consider to be easily accessible from a given place, or how to subdivide geography into familiar units. The only way to discover how to design this application was to actually visit companies in Europe in order to understand what would make the application "work" in their contexts.

We also have conducted many ethnographic studies to evaluate the use of Internet applications in homes in the U.S., Canada, and Latin America. We have observed distinct patterns of usage as well as different attitudes towards features in the different countries, which of course have significant implications for design, instructions, messaging, and even product positioning.

Structured usability evaluations

In a usability evaluation, a representative user performs a set of tasks with a product or Web site (or a prototype of a product or Web site) in order to identify problems in the design. The purpose is to identify aspects of the design that may be failing to cue the user effectively. The focus is on the user's behavior and reactions, such as the path she/he takes through a software interface while trying to accomplish the task. While there is little traditional interviewing during structured usability evaluations, users are often asked to think aloud while they are working in order to gain insights into their thought process and perceptions of the cues available to them in the interface.

Structured usability evaluations are typically conducted in a highly structured manner, using pre-determined task scenarios. These scenarios can be constructed

Figure 6. Observing from behind the mirror in a usability evaluation. Indiana University — School of Library and Information Science, http://www.slis.indiana.edu (photo by Margaret B. Swan).

to represent typical or common user tasks, or they can be designed to probe particular aspects of the design.

Often, these evaluations are performed in a usability lab equipped with a one-way mirror that allows the design team to observe the user's interaction with the product from behind a one-way mirror (see Figure 6). Seeing the user's "pain" firsthand often has a powerful impact on designers and motivates them to make changes. As a result, usability labs serve both a technical and an organizational role: they result in changes to a particular product *and* they sensitize designers to the users, often shifting their attitudes towards UCD. In this way, usability labs can be a very valuable tool for design teams.

In one usability study, we evaluated a building control system whose interface was based on a touch screen display. Because the touch screen display was controlled by a computer in the observation room behind the mirror, we were able to work with the designers to experiment with changes in the menu structure of their product in real time. This resulted in many specific changes to the design approach at the conceptual, logical and physical design levels.

The variety of things which can be tested in a usability lab is limited only by one's imagination and can include:[5]

- Specific elements of product design, such as a navigation pathway, a specific label on a button, or a handle on a physical product
- Out-of-the-box experiences (OOBE), in which people take a product that is packed the way it is (or will be) and set it up while being observed. This might involve the use of a peripheral like a printer or fax machine, a software product like a new computer game or office application, or a consumer product, such as a child's bicycle or furniture
- Web sites and intranet sites
- Interactive processes, such as software downloading or online collaborative gaming
- Documentation and help systems
- Translations
- Low-fidelity prototypes, such as paper prototypes of a navigation pathway for a software product, or a clay model of a hardware product
- High-fidelity prototypes, such as an HTML or Visual Basic™ prototype of software screens or a working model of a hardware product such as a printer or digital camera
- Competitors' products
- Previous releases of a product to be redesigned
- Packaging, to see how easily people can actually get the product out of the packaging

Naturalistic usability evaluations

Naturalistic usability evaluations are a powerful combination of ethnography and usability evaluation. Like ethnographies, they take place in the user's own context (work, home, school, etc.). As in structured usability evaluations, the observers watch for problems, challenges and frustrations. However, unlike structured usability studies, naturalistic usability evaluations are based on the user's *own* tasks, goals, or tools instead of simulated or "typical" tasks, assumed goals and provided tools. These evaluations yield insight into how users think about and actually carry out their own tasks and what their goals are. While ethnography is more focused on understanding fundamental patterns of user behavior and dynamics, naturalistic usability shifts the emphasis toward evaluating specific designs. However, unlike structured usability, it does so in a way that yields information about how the product or software fits in the users' context.

We use naturalistic usability evaluations extensively in our work. One product, the Tablet PC, serves as an excellent example of the power of this type of evaluation. We conducted a series of extended field trials of early prototypes of the Tablet PC (similar to the "Slate" form shown in Figure 7) with the Microsoft team that

Figure 7. "Slate" form of the Tablet PC, similar to that used in the field trials.

was developing this new technology. Participants used these prototypes at work instead of their normal computer for between two and six weeks. In addition to our ethnographic research, during which we observed participants as they used the Tablet in their daily life, we also conducted numerous naturalistic usability evaluations to evaluate specific aspects of the Tablet's design.

As a result of this work, major changes were made to the design of key functions, and some functionality that had proved confusing was discarded. Although Microsoft was also conducting numerous structured usability evaluations in their own usability labs, they did not identify many of the issues that became obvious when the Tablet PC was actually being used in an extended work context. This context made the naturalistic usability evaluations far more powerful than the more limited studies in the usability lab. In addition, we provided data that led to changes in the value propositions and helped shape the positioning of the product.

Figure 8. "Convertible" form of the Tablet PC.
© Hewlett-Packard Company.

These came from seeing how people adapted the Tablet to their own business needs, and the very creative uses that people found for it. Finally, the study identified the need for a new form factor, the convertible (shown in Figure 9) as a key to success for a major consumer sector.[6] Interestingly, the convertible models accounted for almost 80% of Tablet PC shipped in the second quarter of 2004, almost double that of the comparable quarter in 2003 (McArdle 2004). Without the Tablet PC field trials and the naturalistic usability evaluations that were part of them, this design might never have seen the light of day, and the corresponding strong sales might never have materialized.

Table 4 includes a high-level summary of the three primary UCD methods.

Table 4. Summary of the three primary UCD methods.

UCD technique	Characteristics
Ethnographic or Field Studies	• Trained observer visits users "in context" (home, work, school, etc.) • Observer uses a "focus structure" as the skeleton to structure and guide process of observation and questioning • Yield rich data on users, their needs, goals, behaviors, cultural and contextual factors, which may potentially influence design decisions • Especially powerful early in the product planning and design process • Particularly useful for international products, since teams are less likely to understand critical contextual variables • Often result in innovations and/or major product changes

Structured Usability Evaluations	• Representative users carry out simulated tasks while being observed in a usability lab
	• Focus can be broad or targeted
	• Often involve "think-aloud" exercises and pre-determined probing questions
	• Range from informal to formal, quantitative to qualitative
	• Provide insight into the fit of a product with users' mental models and help identify product features that users find confusing, hidden, or difficult to use
	• Should result in specific design recommendations
	• When done cross-culturally (in multiple countries), they also require:
	  ▪ Local recruiting and language facilitation
	  ▪ Localized tasks and process
	  ▪ Oversight by a trained usability person to ensure interpretability
	• Often result in major changes in design at all levels

Naturalistic Usability Evaluations	• Combine ethnography and usability • Take place in users' own environment(s) • Typically based on users' own tasks, goals, and materials • Can be planned or opportunistic • Yield insights about usefulness and fit as well as usability

International UCD research

Usability evaluations and ethnographies undertaken in several different countries can be particularly useful in the development of products for international use. We have conducted many international usability evaluations and ethnographies in the areas of computer products, consumer electronics, Web sites, and software. Our experiences have helped us identify a number of factors that can make or break international user studies. When conducting international or cross-cultural usability evaluations or ethnographies, it is very important that we adapt the evaluation, as well as the recruiting, to the locale. This typically requires translation and local language facilitation, as well as changes to tasks where necessary to reflect local practices. Recruiting of participants should be handled locally as well.[7]

In one recent structured usability study of the out-of-box-experience (OOBE) and documentation for a new type of computer peripheral, we tested the product with users in Europe and Asia. Like the Infiniium Oscilloscope mentioned previously, this product had been designed using a variety of UCD methods, but had not been tested outside of the U.S. prior to our study. To do this study, usability and localization professionals teamed up to conduct a series of structured usability evaluations, which resulted in major changes to the design, layout, and wording of the documentation, translation, and even the packaging. For instance, as a result of these tests, the format of the documentation was completely changed to be seen as being more accessible and "attractive" in Asia, and to be laid out in an entirely different fashion for Europe. These new versions represented significant departures from the preferred form of documentation found in similar usability evaluations in the United States. In addition, because we observed dramatically different set-up behaviors in Asia — Asian users set up the device on the floor, rather than on a desk or table as in Europe and the U.S. — we also made recommendations for changes to the form factor in future versions of the product.

Another structured usability study of a repair kit for a printer involved over 100 participants in 8 countries in Europe and Asia. We observed many difficulties that resulted in restructuring of the conceptual and logical designs of the product, as well as specific illustrations, wordings and translations that were problems for users in different markets. As a result of this study, major changes were made to both the hardware and the software of the kit, along with the packaging, instructions and distribution planning. The final product was very successful.

Along with international structured usability evaluations, we have also conducted many ethnographic studies internationally. In addition to the work done on the Infiniium Oscilloscope and the mapping application described above, we have also conducted visits to home users, home office/small office users, and business users of printers and other peripherals, as well as studies of Web site usage, software application usage, and numerous other products. In one such study, we visited families in the U.S. and in Europe to understand how best to design a new computer peripheral for family use. Not surprisingly, we found that the differences between families (including differences in the ways that families interacted with each other, the location of the computer in the home, patterns of family life, leisure activities, etc.) in different areas of the world were reflected in some different computer usage patterns, and, consequently, in some different future product needs. However, overall, the differences between families that we observed were smaller than those between small office/home office users in a subsequent study of the same type for a related product. We ended up doing far more of the research for that study in Europe than in the U.S., because we found that regional differences in work practices were more pronounced in Europe than in the U.S., where work practices were more related to industry segment than to geography. For instance, users in Dutch small offices/home offices structured their personal work, tended to keep very different working hours, parceled out the work differently amongst members of the office, and even did their tasks differently from users in similar industries in France or Germany.

Naturalistic usability evaluations are particularly well suited to uncovering important information to guide the design of products for international markets. As with ethnographies and structured usability evaluations, we have performed international naturalistic usability evaluations in a wide variety of contexts and for a wide variety of products and services. For instance, in a study of backup devices used in large and small business information technology (IT) departments in Europe, we observed as IT staff prepared to do backups, and then had them do naturalistic usability evaluations of the backup devices they were using to help us understand the current issues and problems with those devices. This was extremely helpful to the design team since we were visiting companies that used the team's product as well as companies that used other brands, including some companies that used a mix of brands. Not surprisingly, there were some significant problems in companies that had a mix of brands. By observing the processes and the problems in context, the team identified some ways to streamline the backup process both for companies that use only their brand, and for those that use multiple brands of backup devices.

In another study, we did naturalistic usability evaluations and observed the problems that users experienced in downloading software from the Internet in several countries in Latin America. We saw some similarities across the countries we visited, but also identified some important country-specific issues which changed the way that downloads were described and introduced in one of the countries we studied.

Conclusion

Localization professionals and UCD professionals are natural allies, since we both share a concern for "ease of use" and "fit" for the users of the products on which we work. Both UCD and localization also make good business sense. Therefore, we can, and must, find ways to support each other by sharing our perspectives, our tools, and our approaches. If we can partner with each other, we can cross-promote our respective disciplines, and, in so doing, increase the likelihood that the users — especially international users — will be considered and, indeed, will drive the design of products and services. The combination of cross-cultural fluency and UCD skills is a powerful one — one which can wield great influence in design teams, and, through the products we work on, in the wider world of product development. While this article has only scratched the surface and introduced the reader to UCD, we urge readers to identify ways that they can work collaboratively to make the users' lives better — through both good design and good localization.

Notes

1. For more on the differences between usability and market research, see Siegel and Dray 2001.

2. For an example taken from software user interface design, see The Daily WTF (http://thedailywtf.com/forums/22267/ShowPost.aspx). For a more systemic view and numerous examples, see the UI Hall of Shame (http://tutor.petech.ac.za/rbotha/UIF4001/HallOfShame.pdf).

3. For a fuller description of mental models, see McDaniel 2003, Johnson-Laird, et al. 1998, or Sasse's excellent dissertation on the subject (1997).

4. For more information on other UCD tools, the interested reader can find more information in a variety of locations, both on the Web (see the IBM, STC, and UPA Web sites, or the excellent compendium of UCD links by Perlman, 2004) and in print (from the classic book by Norman, 1988 to those by Dumas and Redish 1999, Krug 2000, Barnum 2002, Preece, et al. 2002, and Kuniavsky 2003, among others).

5. For more information and links to Usability sites, see: http://www.dray.com/links.html.

6. For more information on the Tablet PC field trials, see Dray, et al. 2002.

7. For an extensive discussion about how to plan for and conduct international user studies, see Dray and Siegel 2005.

References

Barnum, C. 2002. *Usability Testing and Research.* New York: Longman.

Bias, R. and Mayhew, D. (eds). 1994. *Cost-Justifying Usability.* Boston: Academic Press.

———. 2005. *Cost-Justifying Usability: An Update for the Internet Age.* 2nd ed. San Francisco: Morgan Kaufmann Publishers.

Cooper, A. 1999. *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore The Sanity.* Indianapolis, IN: SAMS Publishing Co.

Dray, S. 2004. "Usable in New York, usable in Nairobi: The importance of usability evaluation for international products and Web sites." *MultiLingual Computing & Technology* 15 (5): 31–32.

Dray, S. and Rowland, L. 1998. "Round the world in 18 days: Learnings from an international usability tour." In *The Politics of Usability*, L. Trenner and J. Bawa (eds), 183–190. New York: Springer.

Dray, S. and Siegel, D. 2005. " 'Sunday in Shanghai, Monday in Madrid?!': Key issues and decisions in planning international user studies." In *Usability and Internationalization of Information Technology,* N. Aykin (ed), 189–212. Mahwah, NJ: Lawrence Erlbaum.

Dray, S., Siegel, D., Feldman, E. and Potenza, M. 2002. "Why do version 1.0 and not release it? Conducting field trials of the Tablet PC." *interactions* 9 (2): 11–16. http://www.dray.com/articles.html.

Dumas, J. and Redish, J. 1999. *A Practical Guide to Usability Testing.* Portland, OR: Intellect.

IBM. 2004. "Cost justifying ease of use." IBM Ease of Use. http://www-3.ibm.com/ibm/easy/eou_ext.nsf/publish/23

———. 2004. "Ease of use." *IBM Ease of Use*. http://www-3.ibm.com/ibm/easy/eou_ext.nsf/publish/558

Johnson-Laird, P.N., Girotto, V. and Legrenzi, P. 1998. "Mental models: A gentle guide for outsiders." University of Michigan Interdisciplinary Committee on Organizational Studies. http://www.si.umich.edu/ICOS/gentleintro.html

Krug, S. 2000. *Don't Make Me Think: A Common Sense Approach to Web Usability.* Indianapolis, IN: New Riders.

Kuniavsky, M. 2003. *Observing the User Experience.* San Francisco: Morgan Kaufmann.

McArdle, D. 2004. "Tablet PC market still struggling." *ElectricNews.net*, August 9. http://www.enn.ie/news.html?code=9546693

McDaniel, S. 2003. "What's your idea of a mental model?" Boxes and Arrows Weblog, February 10. http://www.boxesandarrows.com/archives/whats_your_idea_of_a_mental_model.php

Norman, D. 1988. *The Design of Everyday Things.* New York: Doubleday.

Perlman, G. 2004. "HCI bibliography: Human–Computer Interaction resources." http://www.hcibib.org/

Preece, J., Rogers, Y. and Sharp, H. 2002. *Interaction Design: Beyond Human–Computer Interaction.* New York: Wiley & Sons.

Sasse, M.A. 1997. *Eliciting and Describing Users' Models of Computer Systems.* PhD Dissertation, University of Birmingham (UK). http://www.cs.ucl.ac.uk/staff/a.sasse/thesis/Contents.html

Siegel, D. and Dray, S. 2001. "New kid on the block: Marketing organizations and interaction design." *interactions* 8 (2) 19–24. http://www.dray.com/articles.html

———. 2005. "Making the business case for international user centered design." In *Cost Justifying Usability: Revised Edition for the Internet Age*, R. Bias and D. Mayhew (eds), 317–357. San Francisco: Morgan Kaufmann.

Society for Technical Communication Special Interest Group on Usability. http://www.stcsig.org/usability/

Usability Professionals Association. http://www.upassoc.org/

Corpus enhancement and computer-assisted localization and translation

Gregory M. Shreve

Localization, and its constituent process of translation, is costly and labor-intensive because it involves significant and highly skilled human effort. Consequently, a major challenge facing global businesses today is how to make localization and translation faster, cheaper, and of higher and more consistent quality. Technical and business strategies and processes for solving this problem are collectively called internationalization. Most current language industry approaches to internationalization have involved intervention in the document cycle, with an eye to optimizing document processes such as authoring, or in the case of software localization, re-engineering the product itself to make localization easier. The re-engineering of electronic documents and software interfaces and the optimization of "upstream" document cycle processes such as authoring are just two of many possible approaches to internationalization. A concrete example of the latter kind of internationalization is the use of controlled languages to optimize texts that are intended for translation, as for instance the KANT (Knowledge-based, Accurate Natural- language Translation) controlled language described by Mitamura (1999). Developing guidelines for multinational writing or "writing for translation" has long been a part of internationalization strategy in many companies (Adams et al. 1999).

As a precursor to localization, internationalization seeks to reduce the effort and cost associated with localization, while simultaneously increasing its speed and accuracy. From a purely business perspective, the biggest problems that internationalization needs to solve for the language industry are the twin issues of cost reduction and profit enhancement. Because translation and localization are labor- and time-intensive activities, improvement in profit margins has depended primarily on three factors: the development of internationalization techniques, the implementation of business processes tailored for the language industry (language project management, workflow control, translation/localization quality assurance), and the application of translation technology (translation memories, alignment tools, terminology managers, localization applications).

A major internationalization strategy in the language industry, adopted from the software engineering notion of reusability (the same software code can be developed once and then reused in many different applications), is the idea of *language reuse* (Radev 1999; Clough 2001). Language reuse (also: text reuse, linguistic reuse) is the discovery of reusable text and its subsequent annotation and reuse in similar contexts. In the language industry, language reuse has been primarily confined to *translation reuse*, a technique enabled by translation technology, i.e., special software applications called translation memory applications and terminology managers. Once a translator makes a decision about the translation of a particular source text segment, typically a sentence, phrase or clause, a translation memory application aligns the source text translation unit with its target text equivalent and stores the aligned pair in a special database. The translator can then retrieve and reuse the translation should a source-language segment in a new document match a stored language segment. Similarly, terminology managers capture solutions to multilingual terminology problems and store those solutions, with their accompanying documentation, in electronic glossaries. Both of these technologies explicitly acknowledge and advance the principle of reusability. Their purpose is to reduce, for any given translation task, the number of completely new sentences and technical terms to be translated. In all cases, where a previously translated sentence or terminology equivalent can be recalled and used, money is saved. Translation reuse works as a cost-saving approach because the assumption is that the document corpus of most organizations (or at least that part of it relevant to globalization) grows only incrementally, by adding limited amounts of new linguistic material to larger bodies of existing linguistic material. Translation memories and terminology managers store translation content for reuse, thereby *leveraging* previously written or translated materials and reducing the necessity for expensive new authoring or translation effort.

Translation reuse examined

Translation reuse via translation memories and terminology management has played a critical role in reducing the business cost of localization and translation, but there are some restrictions inherent in the reuse paradigm. Translation reuse employs computer-assisted techniques to accumulate and store the results of what is fundamentally a human translation and terminology research effort. Both the terminology databases and the translation memories used by translators as part of computer-assisted translation workstations are necessarily populated by the actions of the translators themselves. As human translators solve terminological or translation problems by dint of research involving parallel texts from the Internet or other document corpora, they create records of those solutions and store them. Over time, as other problems are solved, terminology databases and translation memories are populated with many potential translations for the technical terms and unusual sentences that are often encountered in specialized translation and software localization. These solutions can then be reused in the context of translation workstation software. Although there is an accumulation of translation and terminological data over time, there is a time lag between the advent of any given translation project and the point when translation databases for the project reach an optimal size and scope.

Time lag is not the only problem with the translation reuse paradigm. There are usually also significant restrictions on the scope of records in translation databases. The scope of a record, defined as the number of texts and contexts consulted before making and recording a translation decision, is significantly constrained by the time available for translation research. Due to the constant pressure of deadlines, translators and localizers typically pursue the identification and documentation of terminological or translation equivalents by consulting parallel texts (documents of the same text type) and background texts (documents in the same domain) only to the point where they are satisfied that they have found an acceptable equivalent. Deadlines do not motivate searching additional parallel or background texts for other possible equivalents that may be better or more accurate matches for a source-language term or phrase. The translation quality benefits of extended research may be outweighed by the economic liabilities of extended project time. Of course, this assumes that resources for doing extended terminological or translation research, in the form of collections of parallel and background texts, are readily available and in a useful form. Fundamentally, the basic dependence on human effort to populate terminology glossaries and translation memories has several inherent constraints: the time required to populate databases, economic or other restrictions on availability, research time, and the scope or range of documents consulted to solve terminology and translation problems.

Current business policy in the language industry dictates that localization and translation vendors retain and aggregate the terminology databases and translation memories accumulated by their translators and localizers, creating a shared pool of language resources. As a translation company continues to populate and aggregate its databases in the domains in which it translates, the time lag and human effort for any given subject domain declines, while the range of coverage increases. However, as new domains are added to the translation portfolio (a phenomenon called *domain shift*), the lag / scope / effort problem will reoccur. Even assuming a retention and accumulation policy, the number of parallel texts and research documents consulted before translators make and record translation decisions will still be dependent on access to corpora with relevant documents (the Internet, document repositories, digital libraries) and, of course, project deadlines. A database or translation memory accumulated by pooling the resources of several translators is larger than one compiled by a single individual, but individual records or alignments are still constrained by the same limitations as those of smaller databases.

Some critics have questioned the fundamental presumption of translation reusability as it is currently implemented, arguing that translation memories are really most effective only with documents that change very little over time and have significant sentence repetition. Webb (1999) studied eleven kinds of translation and determined that only four of the eleven types, legal, scientific, technical and commercial, would benefit greatly from translation memory. As soon as variability between documents and document versions is introduced as a factor, the utility of translation memory declines because sentence repetition declines.

Another severe limitation of translation memory is a dependence on the sentence as the primary linguistic translation unit. This dependence has several implications. Macklovitch and Russell (2002) have been critical of the inability of current systems to exploit reusable text at the subsentential level, which we might call microreuse. The authors claim that the great bulk of reusable material consists of elements at the subsentence level: phrases, collocations and multiword terms. Computer-assisted translation systems with finer linguistic granularity could exploit linguistic resources contained within translated sentences. Conversely, Macklovitch and Russell (2000: 137) also criticize the artificial segmentation of the text into discrete, semiautonomous units, and the subsequent loss of access to suprasentential relations, arguing that the "very notion of a document is lost. Not only are the segmented units in a new text extracted from their context and submitted to the database in isolation, but the contents of the database are also stored as isolated sentences, with no indication of their place in the original document." This is an extremely important point. The adaptations that translators and localizers make to documents during translation are not confined to the sentence, but often cross sentence and paragraph boundaries. Linguistic elements considered during translation decision-making — or that should be considered — are also almost certainly not confined to the immediate linguistic microcontext of the sentence.

Clearly, translation memory systems do not preclude access to surrounding sentences or the ability to read a paragraph or a document as part of decision-making, but there is a clear predisposition, even channeling, of translator behavior to sentence-level processing (Dragsted 2002; Webb 2000). An unintended side effect of the focus on the sentence in translation memory systems might be an undesirable feedback effect on the translation process, leading translators to translate, perhaps unknowingly, more in the microcontext than they normally would.

Translation reuse and language reuse are not synonymous. In translation reuse only certain portions of translated texts are reused. Language reuse, at least from the perspective of computational linguistics, could also include discovering and reusing both subsentential and suprasentential textual elements. A broader application of language reuse would also remove the restriction that all reusable (or usable) elements be derived from the relatively limited corpus of translated texts. It is clear that the full potential of language reuse has not been exploited by the language industry.

A corpus-based approach to internationalization

Human terminology and translation research uses source- and target-language parallel texts and background documents that exist in a variety of available machine-readable corpora. These corpora are, for the most part, under used and have not been effectively integrated into translation technology or into the internationalization strategies of most companies. Only some of the results of translation and terminology research, the translated sentence or terminological equivalent, have been stored and reused. Other textual and linguistic objects that could be derived from full texts, such as term contexts, changes made at the suprasentential level by translators or editors, as well as potentially reusable elements that are not translation memory segments, are not retained in translation databases. Although terminology management and translation memories have provided computer assistance for recording some of the results of translation research, there has been precious little assistance for more extensive translation research and resource discovery processes. This assistance is impossible to provide without exploiting the resources of document corpora and the analytic tools of corpus linguistics.

The language industry needs new approaches to internationalization that exploit available corpora to enable automation of some of the laborious human activity involved in supporting translation decisions and populating translation-oriented databases and memories. These approaches should address some of the shortcomings of translation memories and terminology managers by discovering and retaining linguistic, semantic, and textual objects of value to translation in addition to translated sentences and terminology equivalents. Integrating corpora more explicitly in internationalization strategy also means recognizing that human-populated terminology glossaries and translation memories are only the initial applications of translation technology in the language industry.

If relevant corpora could be discovered or constructed and then processed by computational tools, as for example by automatic term extractors, then the translator could be presented with many more terms and term equivalents — in context

— than purely human research would allow. Many more sources of parallel and background texts could be identified and consulted, and many more candidates for term and translation equivalence considered for selection. There would be greater coverage of research materials during the translation process. The use of computational tools would remove restrictions on the range of possible research results exerted by the pressure of project deadlines. Corpus-based methods could substantially shorten the time it takes to populate or fill terminology and translation databases with translation equivalents. All available corpora would need to be leveraged, including the document corpora already owned by organizations, to improve the speed and quality of translation. Where appropriate corpora or appropriately structured corpora don't exist, mechanisms must be developed for creating them.

Translation scholars have already recognized and discussed the importance of corpora in translation studies, translation pedagogy, and translation technology. Mona Baker, in particular, has been a leader in drawing our attention to the potential of corpora and corpus analysis in translation studies (1993, 1995, 1996). Zanettin (1994, 1998) and Varantola (1997), among others, have discussed at length the role that bilingual corpora can play in translating and learning to translate. Some translation technology vendors, mindful of the limitations of the current translation reuse paradigm, are beginning to explicitly integrate bilingual corpora into their translation workbenches (Multicorpora 2002). It is clear that the inclusion of bilingual corpora in computer-assisted translation would overcome some of the limitations of the current generation of workstations.

The prepopulation of terminology databases and translation memories used in the current generation of translation workstations would be one important result of the application of corpus-based methods. Prepopulation refers to the process of automatically providing translation material for the memories and glossaries of translation workstations without relying solely on the actions of translators. However, while the focus in computer-assisted localization/translation is currently on terminology and translation equivalents at the sentence-translation unit level, corpus-based approaches could also allow for the automatic identification and reuse of a wider range of objects useful in localization and translation than is currently offered. These other objects include, but are not limited to: concept systems (also, ontologies, concept catalogues, taxonomies, thesauri), culture-specific document templates derived from the analysis of document structure, collocation sets and phrase collections, reusable document segments and other linguistic or semantic objects. Most of these new linguistic and textual resources cannot be discovered in the limited special-purpose corpora implied by translation memories.

Natural and enhanced corpora

Corpus-based approaches will enable new approaches to internationalization and thus significantly improve the speed, efficiency and accuracy of computer-assisted translation and localization. Using corpora as part of an internationalization strategy implies that they can be manipulated or engineered in some way to be more effective tools for translation and localization. Just as internationalization in software engineering calls for a reengineering of the software kernel of the software applications, using corpora in internationalization implies developing or compiling special-purpose corpora whose contents and linguistic and textual characteristics are compiled, analyzed, and annotated (marked or tagged in some way) in order to make later translation and subsequent authoring faster, more accurate, and more efficient.

Varantola (1997) has referred to specialized corpora created and targeted for a given translation task as *precision corpora*. While she has referred, in the main, to smaller corpora compiled by individual translators, her basic idea could be extrapolated to include the engineering of large-scale corpora on an organization or industry-wide basis specifically to improve translation and localization in specific domains. Both Ahmad et al. and Varantola have referred to the ephemeral quality of the special corpora constructed to assist in translation activity. Ahmad et al. (1994) write of *virtual corpora*, ephemeral constructs created to help the translator complete his or her translation task. Varantola (2000) has also used the interesting phrase *disposable corpora* in the same context. Given the current practice in the language industry of retaining and aggregating all translation resources produced for a company by its translators, it would seem illogical to discard the translation information gathered as a result of translation research or to ignore the potentially relevant information that could be gathered and stored if bilingual corpora could be discovered (or constructed) and exploited. This argues for Ahmad and Varantola's ephemeral corpora to be made permanent and integrated into the translation resources and translation technology of an organization or localization vendor.

Varantola's conception raises a question. How would one begin to compile a precision corpus large enough, organized enough, and comprehensive enough to be useful in computer-assisted translation and viable as an internationalization strategy? One answer would be to begin by analyzing an organization's naturally occurring collection of documents, what we might call a *natural corpus* and then use it as a *seed corpus* to construct a large-scale precision corpus. The natural corpus is not representative of the entire language or textual system, but is, as Noam Chomsky noted so long ago, linguistically skewed (1957: 159). For the purposes at hand, the skewed nature of the corpus is desirable. We are not interested in general language or in discovering the formal characteristics of the language system as a whole, but in those linguistic and textual features that are domain- or language community-bound — special language and text. The natural corpus can be assumed to contain exemplars of the specialized linguistic and textual preferences of a specific and well-defined language community. In some sense, it is a circumscribed text world, a finite repertoire of textual interaction structures used in a particular communicative community (Neubert and Shreve 1992: 41). For the sake of conceptual completeness, we can define a natural corpus as the entire set of documents produced and stored in an organization. An *intranet-bounded natural corpus* is that subset of the natural corpus of an organization which is in machine-readable format and discoverable by computational means.

A corpus-based approach to internationalization would entail analysis of the natural corpus to construct models of its specialized content (domain model) and range of document types (document structure model). These two models reflect the contents (topics, subject areas, or domains, as well as specialized linguistic objects such as terms or phrases) and the kinds of document types of greatest import and utility to the organization. Once constructed, these models can then be used to provide parameters to intelligent agents, such as Web spiders (automated Internet search programs that "crawl" the Web looking for documents), so that they may acquire and integrate new documents into the corpus in a specific, targeted manner from the Internet and/or other document repositories outside the original boundaries of the organization's corpus. The construction of the domain and document structure models is the mechanism for using the natural corpus as the seed for a larger precision corpus. New documents can be added to the original seed corpus if they meet certain criteria — for instance, if the distribution of diagnostic terminology in target documents meets certain thresholds. The new corpus thus constructed could be a significant enhancement over the original corpus, as it can be assumed to contain a more complete set of the prototypical instances of the specialized vocabulary, semantic relations, linguistic usages, phraseology, and document formats and document types that are of greatest import and utility to the organization. This *enhanced corpus* can be taken to more accurately reflect existing practices in the written communications of the linguistic community to which the organization belongs (see Figure 1).

The natural corpus prior to enhancement is typically not annotated; it is a *raw corpus*. As SGML (Standard Generalized Markup Language) and its variant, XML (Extensible Markup Language), become more commonly used in business, natural corpora will contain preexisting annotation. Even if previous annotation exists, it is most likely the case that application-specific annotation will have to be added to the corpus to make it useful for computer-assisted translation and an effective tool for internationalization. This implies tagging the results of localization/translation-specific corpus analysis using metadata expressed in a markup language

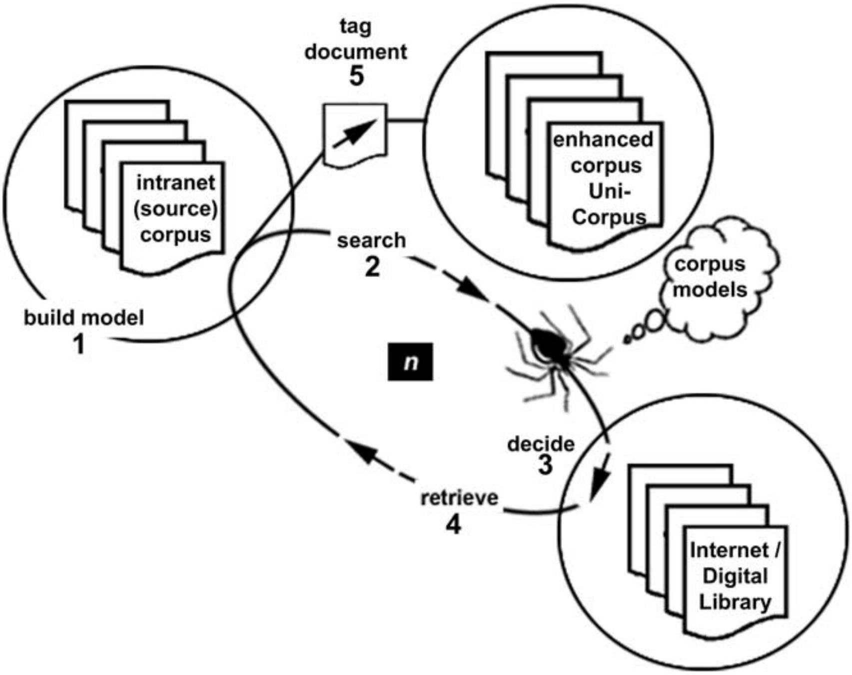

Figure 1. Domain modeling using intelligent agents.

such as XML. Many translation scholars have recognized the utility of markup languages in translation corpora (Luz and Baker 2000).

Markup allows for the description and later retrieval of linguistic, semantic, and textual objects of relevance to translation and localization that are discovered by corpus analysis. Non-linguistic information related to the parameters of the localization or translation task can also be stored. Markup objects are not limited to terminology and aligned translation units. Collocation and phrase collections, term contexts, thesaurus or concept relationships, style and usage patterns, recurrent text segments, or textual superstructures diagnostic of particular textual forms could also be discovered and annotated. Metadata schemes for annotating these elements could be developed or adapted from existing schema. Yves Savourel (2000: 67) and others have also argued for the inclusion of localization information in documents, using a kind of Localization Markup Language. Markup and metadata schema are already in widespread use for translation memories and terminology management (TMX, XLT).

Engineering enhanced corpora

Building enhanced corpora from natural corpora entails the application of specific language engineering processes. The most important processes are intelligent corpus modeling and building, corpus enhancement, and corpus multilingual replication. Once enhanced corpora are constructed, they can be integrated into a new generation of translation workstations, mined for translation or authoring-relevant data, or even shared in large distributed peer-to-peer networks.

Intelligent corpus modeling and building is a process employing intelligent agents such as Web spiders to create an enhanced document corpus. Intelligent corpus building assumes, as discussed earlier, that an intranet-bounded natural corpus represents a model of the text world of an organization. The corpus is a potentially large, but finite, set of exemplars of the document types and contents of greatest interest and concern to the corpus-owning organization. Analysis of this natural model — which is intrinsic and implicit — can yield more explicit models of the document types and linguistic and semantic contents contained within the corpus. These more explicit models can be used to direct the search and analysis of intelligent agents and used to enhance the corpus according to desired parameters. The use of the Internet for corpus construction and the automatic construction of corpora using agents has been described in the literature, especially by Hassel (2001). Crowder and Nicholas (1996) have also proposed architectures for updating text corpora and their associated metadata using agents.

The corpus domain model (see Figure 2) assumes that the textual-linguistic structures of the documents encode translation-relevant content data that can be discovered by computational means. There are several approaches to modeling document content (Jones 1992; Boguraev and Kennedy 1997; Boguraev 1998). One method of particular value to translators would be simply to capture and organize the terms contained in a corpus. The distribution of terms across the natural corpus could be taken to be a linguistic representation of the specialized knowledge structures (domains) of the corpus (Richter 1995). The domain model includes a hypothesis of the range and intersection of corpus domains represented by the terminology, as well as hypotheses regarding the diagnostic criteria for identifying and organizing domains and their constituent concepts into knowledge representations. One process for determining the special vocabulary used in the corpus domain model is term extraction (also: term parsing, term acquisition).

Approaches to term extraction are well described in the literature, and a number of approaches could be used in constructing simple terminology-derived domain models (Church and Dagan 1994; Bourigault 1992; Daille, Gaussier and Lange 1994; Daille 1995; Justeson and Katz 1995). Statistical, natural language processing (NLP) and hybrid statistical/NLP approaches are all feasible, although statistical approaches appear to be the most promising approach for building the large-scale corpora needed by the language industry. Blom (1998), for instance, describes a process that exploits the statistical properties of words most likely to be terms. According to Blom, terms are high frequency content words with a nonrandom Poisson distribution over a given corpus. Using statistical techniques, a list of unigrams (individual words) considered to be term candidates could be compiled and the distribution of the candidate terms over the corpus calculated. Those content words showing a random distribution over the corpus would be removed from the term candidate list and those that show nonrandom distribution would be retained. Blom's process can also be extended to identify two-word terms by determining the collocational potential and bond strength of unigram pairs.

Of course, not all terms are two-word terms. Other procedures can be applied to discover term-like n-grams (multiple word units) by examining the words in their immediate context. One such statistical procedure has been described by Smadja (1993) to identify and extract collocates. A primary objective of identifying collocations is to discover multiple-word terms, but the technique may also be used to identify stereotypical or "boilerplate" language and word associations. The techniques offered by Blom and Smadja are not the only ones that could be used for domain model building; they are offered for the sake of illustrating some promising approaches.

Of course, terms, while extremely important to translators, are not the only kind of structures that can be identified and included in the domain model. Subsentence structures, as for instance phrase collections, could be discovered and stored, as called for earlier by Malkovitch and Russell. Godby (2001) and Sojitra (1998), among others, have described mechanisms for extracting phrases from text corpora. Many researchers (Smadja 1993; Goldman, Nerima and Wehrli 2001; Kenji, Yasuhiko, Takashi and Yoneo 1994) describe both natural language and statistical approaches to collocation extraction.

Many of the term-like structures discovered while modeling the domain can be arranged into concept systems. Concept systems are semantic networks that indicate the meaning relationships between linguistic units. Of particular importance for computer-assisted translation and authoring purposes is the use of concept systems as a mechanism for aggregating multilingual equivalents of terms and monolingual terms that are synonyms into a common concept object. Discrete concept objects are then linked in semantic networks that indicate hierarchic, pragmatic, or other semantic relationships among them. Faber and Sanchez (2001: 192) speak eloquently of the importance of the concept structures in translation, claiming that the "representation of conceptual structures in a specialized domain is an essential part of terminology processing for translators … the organization of concepts in translation-oriented terminology management is a tool to facilitate knowledge acquisition."

The automatic generation of conceptual structures from linguistic resources discovered in a corpus can be accomplished by a number of mechanisms, all of which may be utilized alone or in combination as necessary and appropriate. Latent semantic indexing, for instance, could be used to discover which terms in a corpus appear to be related. Latent semantic indexing is a method that organizes information discovered in text corpora into semantic structures. According to Foltz (2002) the method "takes advantage of some of the implicit higher-order associations of words with text objects. The resulting structure reflects the major associative patterns in the data…" Once concept structures have been identified, they could be presented visually in the form of concept browsers. One of the major problems translators face is lack of domain knowledge, particularly when translating terminologically dense materials. A partial solution to this problem, short of extensive reeducation in the domain, would be to provide assistance in understanding the relationships of terms via what Hoppenbrouwers (1998) has called topic-level concept browsing and terminology navigation. This kind of browsing could be possible if concept structures could be automatically extracted from corpora.

We have just described approaches to identifying and annotating objects such as terms, phrase collections, collocation sets, and conceptual structures. However,

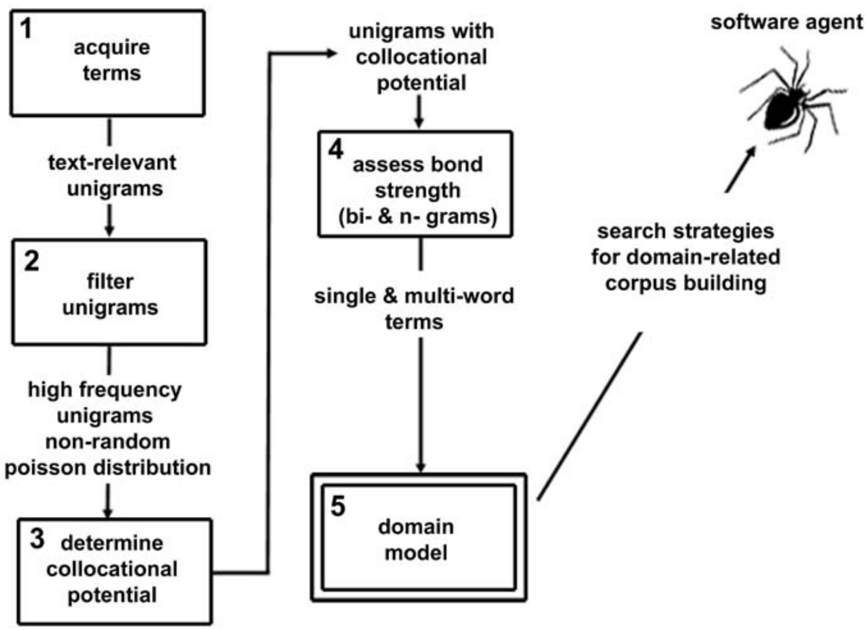

Figure 2. Enhanced document corpus engineering.

the intranet-bounded natural corpus also encodes information about document logical structure and physical layout. Document logical structure reflects cultural norms of document organization and the logical relationships and sequence of document elements. Translation scholars have long recognized intercultural variation in document (textual) structure as an important translation variable (Hatim and Mason 1990; Reiss and Vermeer 1984). Logical structure can be generally decomposed into logical elements such as chapters, sections, subsections, paragraphs, and so on. Physical layout focuses on characteristics of the display medium, such as pages, lines, characters, margins, indentation, fonts, and so on. The relationships of logical structure to physical layout are also culturally determined. The range of options for physical layout will vary, of course, by medium.

Documents have internal textual-linguistic semantic structures that are associated with function and purpose and based in culture-bound forms of social interaction, as for example the culture-bound structure of a lease or contract. Specific patterns of these internal structures (recurrent collocations or phrases, recurrent sentence sequences, patterns of headings and subheadings, diagnostic lexemes) can be taken to be diagnostic of particular document types, as, for instance, technical reports, certain kinds of Web pages, memoranda, patents, contracts, and so on. An intranet-bounded natural corpus is presumed to contain an intrinsic model of the distribution of document types of greatest interest and concern to the corpus-owning organization. A corpus document structure model is a hypothesis of the range of document classes in the corpus and a hypothesis regarding the diagnostic criteria for classifying the documents found in the corpus as to type. The document structure model is a specification of the logical structural entities that occur within the intranet-bounded natural corpus, their hierarchical relationships, and associated physical layout (see Figure 3).

The corpus document structure model has a granularity that ranges from the micro-structural level (diagnostic criteria that reside at the collocation, phrase, and sentence level) to the macro-structural level (diagnostic criteria applying to larger segments of the documents such as paragraphs or groups of paragraphs) or to the super-structural level (titles, headings, and subheadings). To the extent structures at all levels can be determined computationally and described via a metadata schema (such as that provided by the Text Encoding Initiative — see Appendix) using a markup language such as XML or SGML, they can be recorded and preserved for the use of authors and translators. In cases where markup of such documents already exists, for instance where there has been application of style codes, a mapping of existing markup to a standard metadata scheme could be employed.

Computational methods for determining document structure patterns are dependent on the encoding and storage format of the documents to be analyzed. Some systems for document structure identification begin with corpora of scanned images (such as those in many document management systems) and attempt to statistically model document structure by image analysis. Other approaches analyze documents in their native text encoding (Brugger 1997; Brugger, Zramdini and Ingold 1997; Klink, Dengel and Kieninger 2000). When discovered during parsing and analysis, constituent elements (titles, headings, sections, subsections, paragraphs, list items) could be tagged and their corresponding physical characteristics, where present, extracted and stored.

When analysis is complete, the logical description of a document can be extracted from the document and presented as a tree structure (with the entire document as the root node and individual constituents as leaf nodes). Any individual constituent element can be extracted and compared to similar constituents in other documents. Constituents from many documents can be compared and recurrent patterns recorded, creating the possibility of developing prototypical or classificatory properties for document constituents and document classes.

Gommlich and Förster (1991) have argued for the value of what they call "text patterns" for translators. According to Gommlich (1995: 221), text patterns include "both information about the sequential and hierarchical structure of types of texts in a specific source or target language and information about text-type

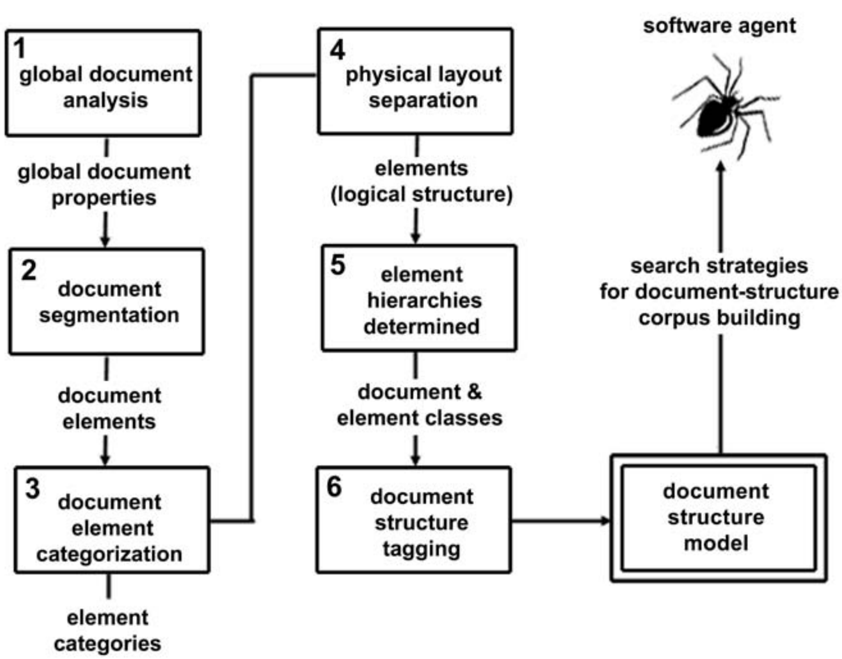

Figure 3. Document structure modeling.

specific translation cues." Using enhanced corpora it is possible to extract the text patterns Gommlich calls for, providing, for the first time, a translation resource that addresses cultural variability in document and text logical organization and physical layout. For the first time, the compilation, analysis, and reuse of collections of parallel texts can be included in computer-assisted translation.

Adding multilingual documents to the enhanced corpus

After the natural (seed) corpus has been enhanced by annotation and the addition of targeted documents, it is still monolingual. Clearly, for the purposes of translation and localization, a mechanism for discovering, acquiring and integrating multilingual documents into the corpus is necessary. If multilingual resources are added, they can themselves be analyzed and tagged so as to allow for the cross-linguistic alignment of linguistic and textual resources and to provide information on culture-bound preferences with respect to the structure and format of documents. If the enhanced corpus remains monolingual, it may still serve useful purposes in terminology standardization efforts, controlled language initiatives, and as part of workstations for computer-assisted authoring of technical or other specialized documents.

As was the case earlier, intelligent agents can be deployed on the Internet or in other document repositories to search for target-language documents with particular characteristics. Agents can also be used to couple elements of the monolingual corpus (for instance, terms, phrases, collocates) with their potential multilingual equivalents or analogues, thereby replicating the monolingual corpus across languages. Corpus replication is a process whereby source-language documents and document elements in the modeled monolingual corpus are matched with comparable target-language documents or elements using methods based in computational corpus linguistics.

One of several methods that could be used for this replication is the so-called comparable context method described by Peters and Picchi (1996, 1997). This is not the only approach possible, but is described here for the purpose of illustration. Of course, any existing translations of documents within the original intranet-bound corpus should be located and exploited. Sets of translated documents on the Internet should also be discovered and used if they fall within the parameters given to the intelligent agent.

Given the scarcity of naturally occurring parallel corpora, most often corpus replication will proceed by searching the Internet and other document repositories for multilingual documents that are not translations. These foreign language documents are retrieved and annexed to the original corpus by intelligent agents if they are determined to be within the same domain space as the modeled monolingual corpus, or if they fall within the compass of the document types in that corpus. Once retrieved and annexed, they are themselves modeled with reference to document structure and domain to reveal any culture-bound differences in document structure and domain.

The replication process begins by using the domain model of the enhanced monolingual corpus to construct a precision L_2 *comparable corpus*. Zannetin (1998) and others have emphasized the importance of manually constructed comparable corpora in translator training. A comparable corpus is composed of documents in a foreign language that are not translations of the source-language corpus but are in the same domain. Existing approaches to the automatic extraction of multilingual terminology from a multilingual document corpus depend primarily on translation alignment of the translation units (typically sentences) between documents that are translations of one another, so-called *parallel corpora*. Such corpora are not very common and usually only exist as the output of human translation activity (Smadja and McKeown 1994). Successful corpus replication depends on methods for automatically discovering multilingual equivalents for source-language terms, collocations, or phrases that do not depend on aligned parallel corpora. There is significant current research in using comparable corpora in translation (Fung 1995; Masuichi et al. 2000; Peters and Picchi 1997; Rapp 1999) that could be adapted for the creation of precision comparable corpora.

For example, using the comparable context method of Peters and Picchi, the terminology extracted during the construction of the largely monolingual corpus domain model during intelligent corpus building could be used as the basis for building the L_2 comparable corpus (see Figure 4). The significant source-language terms, phrases and collocations identified in the monolingual phase of corpus building become a means to bootstrap the search for foreign language documents falling within the same domain as the original documents. In the first stage of the replication process, a general language bilingual machine dictionary for each of the target-languages of the replication process is used to provide lexical equivalents for as many of the contexts of the terms as possible. These contexts are available in the previously constructed L_1 domain model. Combinations of translated words and phrases are then used as search strategies to locate and retrieve documents where there is a significant copresence of the lexically translated target-language words in a target-language context. Significant copresence is based on statistical assessment of the probability that sets of co-occurring L_2 words represent lexically equivalent contexts for a given L_1 set of words. Lexical translation of L_1 words and expressions does not yield actual translation equivalents. It is an axiom in terminology studies that most single- and multiple-word terms cannot be accurately translated using general language dictionaries. The use of lexical-level machine translations in the

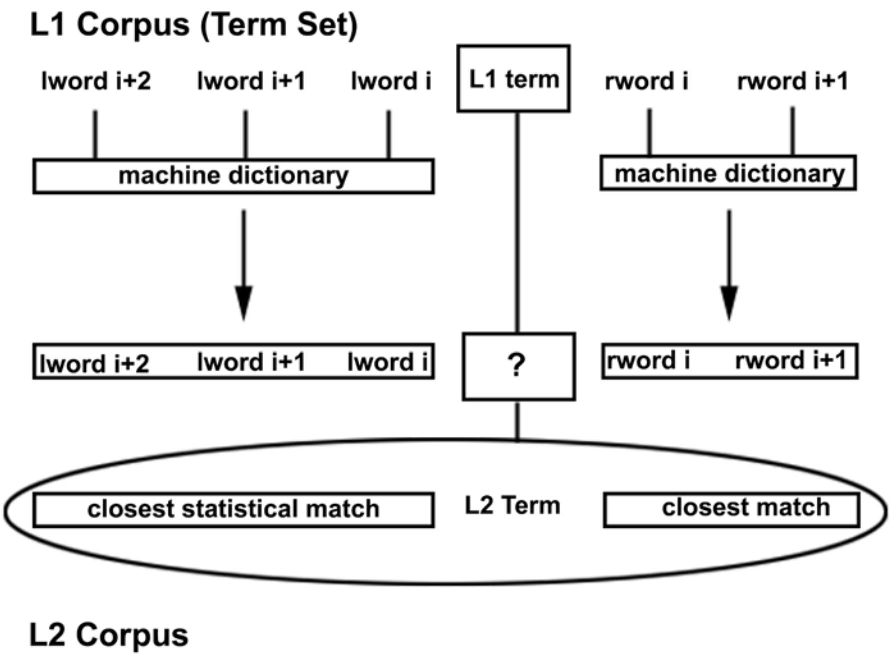

Figure 4. Multilingual corpus replication.

technique described here is to provide a bootstrap to start a search for domain-equivalent target-language documents.

The accuracy of the replication process could be enhanced in several ways, including seeding with L_2 terms derived from an existing machine-readable bilingual terminology. Another approach would be to analyze a select set of terminologically dense L_2 texts in the proper domain and use the resulting set of terms and expressions as the search strategy for agents to retrieve further target-language documents. If documents in parallel corpora (documents that are translations of one another) are found or are available, they can be used to provide an initial set of L_2 terms for bootstrapping the multilingual search. The comparable context procedure summarized here could operate without using standard terminologies or seed documents. Stand-alone operation would be required in situations where a domain and its representative documents are relatively new and standard terminology glossaries or seed texts are not yet available.

The originally monolingual corpus is partitioned as multilingual candidate documents are discovered and retrieved by the agents. The multilingual documents added by the cloning process compose new partitions, one partition for each new language added. As the number of candidate documents added to new multilingual partitions increases, the partitions can be analyzed in the same fashion as described earlier, resulting in sets of comparable L_2 terms, collocations, phrases, or concept structures. The L_1 and L_2 sets can be compared and the linguistic resources in the L_2 ranked as to the probability that they are equivalents or analogues for resources in the L_1. Candidacy can be further validated by human review, against parallel corpora, or against standard terminologies. Indeed, the general strategy will be to generate candidate equivalents and seek validation continuously during the operation of the translation or authoring context in which the resources are used. If the intelligent agent can refresh its search strategies by using validated parameters, this ensures that the agent becomes more intelligent in its replication behavior as the size of the multilingual portion of the corpus increases. This would require provisions for iteratively modeling the multilingual partition as it is being constructed and for improving confidence in the equivalency candidates identified by purely automatic means.

It has long been a staple principle of translation studies that document or textual structure is culturally bound (Neubert and Shreve 1992). The corpus document structure model determined for the original, monolingual enhanced corpus is valid only for the culture that produced the documents on which it was based. To produce models of document structure valid for other cultures, the original monolingual enhanced corpus document structure model must also be cross-culturally replicated. The multilingual replication of the original corpus domain model contains a bootstrapping problem, a requirement for generating initial search parameters to allow an intelligent agent to find and retrieve an initial set of relevant L_2 documents from the Internet or other document repository. It would appear at first glance that a similar bootstrapping problem does not exist with respect to the multilingual replication of the corpus document structure model, since the replication of the corpus domain model has *de facto* created an initial L_2 document set. (This implies that domain modeling is done first, followed by document structure modeling.) The set of L_2 documents generated by domain modeling can be used as the catalyst for beginning the multilingual replication of the corpus document structure model. The initial L_2 document set would be analyzed as described earlier and document logical structure and physical layout determined.

However, this approach raises the issue of *isomorphism*. For each set of terms, phrases, and collocations generated for the L_1 corpus, the objective is to generate at least one or more potentially valid equivalent sets in the L_2 corpus. The replicated corpus is roughly isomorphic with the original in terms of size and domain scope. However, using the L_2 corpus generated by the cloning of the corpus domain model as a bootstrap does not guarantee that a corpus document structure model isomorphic to that generated for the L_1 can be replicated for the L_2. There is no guarantee that the bootstrap corpus contains a range of document types equivalent to that of the original monolingual corpus structure model even if it covers the same domains. The goal in this second stage of replication is not to create a comparable corpus (a set of texts in the same domain) but a set of texts of the same text type. In translation studies such texts are called parallel texts. However, in corpus linguistics the use of the word "parallel" in the term "parallel corpus" diverges from its use in translation studies. A corpus of parallel texts (in the translation studies sense) would be texts of the same type, fulfilling similar interactional aims and with isomorphic or analogous organization and structure. Such corpora could be termed *isomorphic corpora*. The automatic discovery and analysis of cross-cultural differences and similarities in document types between two languages would be a significant benefit of integrating enhanced corpora into computer-assisted translation, so the ability to construct corpora of parallel texts is extremely important.

Solving the problem of isomorphism will require searching for L_2 documents matching key diagnostic criteria for document classes discovered during the construction of the L_1 document structure model. Once the initial L_1 document structure model has been determined, key indicators could be extracted and used in the development of a replication heuristic. For instance, once it has been determined that one of the diagnostic properties of document class *memorandum* is the appearance of standard text segments (TO, FROM, DATE, SUBJECT), a document replication heuristic can be used to search for L_2 documents having comparable functionally if not linguistically equivalent indicators. Documents retrieved can be validated against other L_1 document-derived heuristics (for instance, patterns of length, terminological density, appearance of expected standard collocations, and other indicators as described earlier). Documents whose diagnostic criteria most closely match across languages will be assumed to belong to equivalent document classes.

Conclusion

Once the multilingual replication process is completed, an organization will own a large precision corpus that has been organized and annotated with authoring, translation, and localization in mind. The corpus will exist in two or more language partitions with cross-language and cross-cultural alignment and/or comparison among linguistic, conceptual, and textual units possible. The corpus has been constructed in such a way as to allow for the efficient discovery of reusable translation and localization resources. These resources, once discovered, are recorded and organized into more complex structures by the use of specialized markup. We have termed this new corpus an enhanced corpus.

The enhanced corpus can be exploited for resource discovery. Text mining, a technique well studied in the literature, is the process of analyzing a body of texts to extract information from them for particular purposes (Hearst 1999; Soderland 1997). Within the context of the current discussion, we could speak more precisely about corpus mining. Given the control of the way the enhanced corpus was constructed and the extent of directed analysis and annotation, corpus mining would be more similar to structured data mining. The process of creating the artificially enhanced corpus (and the concomitant creation of the corpus domain model and the corpus document structure model) involves parsing and then annotating any discovered structures, including terms, multiword terms, collocations, standard phrases, logical document elements, and so on, using tags associated with appropriate metadata schemas. As the enhanced corpus accretes during the corpus building and corpus replication stages, all documents that are added, and the elements discovered within them, are analyzed, categorized, and tagged in relation to these schemas. The analysis and tagging process converts an unstructured body of data into a structured body.

The approach described above creates a multilingual enhanced corpus from which multilingual terminologies, phrases, collocations, concept networks, translation alignments, document structures, and other objects serving to make translation and localization easier and more cost effective may be extracted. Enhanced corpora should be integrated into the next generation of authoring and translation workstations to improve the ability of companies engaged in large-scale intercultural communication to deal effectively with the twin issues of localization cost and effort.

Creating enhanced corpora will not be without its problems. Significant issues of copyright and digital rights management, for instance, have not been considered. In addition, there is the issue of determining how to exclude sensitive or protected information from corpora that are to be made available to groups outside of a given organization. Still, it is clear that enhanced corpora are a viable approach to internationalization. Properly used, enhanced corpora will reduce translation and localization, and enhance profitability. The precondition is, of course, the proper engineering of the corpus using the principles described above.

References

Adams, A.H., Austin, G.W. and Taylor, M. 1999. "Developing a resource for multinational writing at Xerox Corporation." *Technical Communication* 46 (2): 249–254.

Ahmad, K., Holmes-Higgin, P. and Abidi, S. 1994. "A description of texts in a corpus: 'Virtual' and 'real' corpora." In *EURALEX 1994 Proceedings*, W. Martin, W. Meijs, M. Moerland, E. ten Pas, P. van Sterkenburg and P. Vossen (eds), 390–402. Amsterdam: Vrije Universiteit.

Baker, M. 1993. "Corpus linguistics and translation studies: Implications and applications." In *Text and Technology: In Honour of John Sinclair*, M. Baker, G. Francis and E. Tognini-Bonelli (eds), 233–250. Amsterdam/Philadelphia: John Benjamins.

———. 1995. "Corpora in translation studies: An overview and some suggestions for future research." *Target* 7: 223–243.

———. 1996. "Corpus-based translation studies — The challenges that lie ahead." In *Terminology, LSP and Translation*, H.L. Somers (ed), 175–186. Amsterdam/Philadelphia: John Benjamins.

Blom, B. 1998. "A statistical and structural approach to extracting collocations likely to be of relevance in relation to an LSP sub-domain text." *The 11th Nordic Conference on Computational Linguistics*. http://www.nodali.sics.se/bibliotek/nodalida/1998_kph/NODA98–19/NODA98–19.html (no longer available)

Boguraev, B. and Kennedy, C. 1997. "Salience-based content characterization of text documents." *Proceedings of ACL'97 Workshop on Intelligent, Scalable Text Summarisation*, 2–9. Madrid: Association for Computational Linguistics.

Boguraev, B., Kennedy, C. and Brawer, S. 1998. "An architecture for content analysis of documents and its use in information and knowledge management tasks." *SIGCHI Bulletin* 30 (2): 64–71.

Bourigault, D. 1992. "Surface grammatical analysis for the extraction of terminological noun phrases." *Proceedings of the 14th International Conference on Computational Linguistics*, 977–981. Nantes: COLING.

Brugger, R. 1997. "A statistical approach to document structure modeling." Informatics Institute of the University of Fribourg. http://diuf.unifr.ch/people/brugger/papers/97_jed/overv_en.html

Brugger, R., Zramdini, A. and Ingold, R. 1997. "Modeling documents for structure recognition using generalized n-grams." *Proceedings of ICDAR'97*, 56–60. Ulm: Fourth International Conference on Document Analysis and Recognition.

Church, K. and Dagan, I. 1994. "Termight: Identifying and translating technical terminology." *Fourth Conference on Applied Language Processing*, 33–40. Stuttgart: Association for Computational Linguistics.

Clough, P. 2001. "Measuring text reuse in a journalistic domain." *Computational Linguistics UK* (4): 53–63.

Crowder, G. and Nicholas, C. 1996. "Using statistical properties of text to create metadata." *First IEEE Metadata Conference*. Silver Springs, MD: IEEE. http://www.computer.org/conferences/ meta96/crowder/onefile.html

Daille, B. 1995. *Combined approach for terminology extraction: Lexical statistics and linguistic filtering*. PhD Dissertation, University of Paris.

Daille, B., Gaussier, E. and Lange, J.M. 1994. "Towards automatic extraction of monolingual and bilingual terminology." *Proceedings of COLING 94*, 515–521. Kyoto: COLING.

Dragsted, B. 2002. *Segmentation in translation and translation memory systems*. Research Project Outline. Copenhagen: Copenhagen Business School Department of English.

Faber, P. and Sánchez, M. 2001. "Codifying conceptual information in descriptive terminology management." *Meta* 46 (1): 192–204.

Foltz, P.W. 2002. "Using latent semantic indexing for information filtering." Boulder, CO: University of Colorado. http://www-psych.nmsu.edu/~pfoltz/cois/filtering-cois.html. Originally published as Foltz, P. W. 1990. "Using Latent Semantic Indexing for Information Filtering." In *Proceedings of the Conference on Office Information Systems*, R.B. Allen (ed), 40–47. Cambridge, MA: N.p.

Fung, P. 1995. “Compiling bilingual lexicon entries from a non-parallel English-Chinese corpus.” *Proceedings of the Third Workshop on Very Large Corpora*, 173–183. Boston, MA: Association for Computational Linguistics.

Godby, J. 2001. “Two techniques for the identification of phrases in full text.” *Journal of Library Administration* 34 (1/2): 57–65.

Goldman, J.-P., Nerima, L. and Wehrli, E. 2001. “Collocation extraction using a syntactic parser.” *39th Annual Meeting and 10th Conference of the European Chapter of the Association for Computational Linguistics (ACL'39)*, 61–66. Toulouse: Université des Sciences Sociales.

Gommlich, K. and Förster, K. 1991. “Textpatterns in a computer-assisted translators' workstation.” *Hermes* (6): 5–30.

———. 1995. “The X-bar principle as an analogon for textpatterns.” In *Basic Issues in Translation Studies*, G. Shreve, A. Neubert and K. Gommlich (eds), 221–230. Kent, OH: Institute for Applied Linguistics.

Hassel, M. 2001. “Internet as corpus: Automatic construction of a Swedish news corpus.” Uppsala: The 13th Nordic Conference on Computational Linguistics. http://www.nada.kth.se/~xmartin/papers/NewsCorpus_NODALIDA01.pdf

Hatim, B. and Mason, I. 1990. *Discourse and the Translator*. London: Longman.

Hearst, M. 1999. “Untangling text data mining.” *Proceedings of ACL'99*, 3–10. College Park, MD: Association for Computational Linguistics.

Hoppenbrouwers, J.J.A.C. 1998. *Analysis and Design Advanced Functionality.* Luxembourg: European Commission. http://www.bib.uab.es/project/eng/d41.pdf

Jones, R.L. 1992. “Automatic document content analysis: The AIDA project.” *Library Hi Tech* (10): 113–117.

Jones, R., McCallum, A., Nigam, K. and Riloff, E. 1999. “Bootstrapping for text learning tasks.” *IJCAI-99 Workshop on Text Mining: Foundations, Techniques and Applications*, 52–63. Stockholm: International Joint Conference on Artificial Intelligence.

Justeson, J. and Katz, S. 1995. “Technical terminology: Some linguistic properties and an algorithm for identification in text.” *Natural Language Engineering* 1 (1): 9–27.

Kenji, K., Yasuhiko, K., Takashi, O. and Yoneo, Y. 1994. “A comparative study of automatic extraction of collocations from corpora: Mutual information vs. cost criteria.” *Journal of Natural Language Processing* 1 (1): 21–33.

Klink, S., Dengel, A. and Kieninger, T. 2000. “Document structure analysis based on layout and textual features.” *Proceedings of the Fourth IAPR International Workshop on Document Analysis Systems*, 99–111. Rio de Janeiro: International Association for Pattern Recognition.

Luz, S. and Baker, M. 2000. “TEC: A toolkit and application program interface for distributed corpus processing.” *Workshop on Web-Based Language Documentation and Description*. Philadelphia, PA: Linguistic Data Consortium. http://www.ldc.upenn.edu/exploration/expl2000/papers/luz/index.html

Macklovitch, E. and Russell, G. 2002. “What's been forgotten in translation memory.” *Proceedings of AMTA 2000*, 137–146. Cuernavaca: Association for Machine Translation in the Americas.

Masuichi, H., Flournoy, R., Kaufmann, S. and Peters, S. 2000. “A bootstrapping method for extracting bilingual text pairs.” *Proceedings of the 18th International Conference on Computational Linguistics*, 1066–1070. Saarbrücken: COLING.

Mitamura, T. 1999. “Controlled language for multilingual machine translation.” *Proceedings of Machine Translation Summit VII*, 46–52. Singapore: N.p.

MultiCorpora R&D Inc. 2002. "The full-text multilingual corpus: Breaking the translation memory bottleneck." Quebec: MultiCorpora R&D Inc. http://www.multicorpora.ca/papers/WhitePaper_1.pdf

Neubert, A. and Shreve, G.M. 1992. *Translation as Text*. Kent, OH/London: Kent State University Press.

Peters, C. and Picchi, E. 1996. "A system for cross-language information retrieval." *ERCIM News Online Edition* 27. http://www.ercim.org/publication/Ercim_News/enw27/peters.html

———. 1997. "Across languages, across cultures: Issues in multilinguality and digital libraries." *D-Lib Magazine* 3 (5). http://www.dlib.org/dlib/may97/peters/05peters.html

Radev, D. 1999. *Language reuse and regeneration: Generating natural language summaries from multiple on-line sources*. PhD Dissertation, Columbia University.

Rapp, R. 1999. "Automatic identification of word translations from unrelated English and German corpora." *Proceedings of the ACL-99*, 519–526. College Park, MD: Association for Computational Linguistics.

Reiss, K. and Vermeer, H. 1984. *Grundlegung einer Allgemeinen Translationstheorie*. Tübingen: Niemeyer.

Richer, M., Schmidt, G. and Schneider, M. 1995. "Terminology and knowledge representation in complex domains." *Proceedings of the Gesellschaft für Klassifikation*, 414–426. Basel: Springer Verlag.

Savourel, Y. 2000. "XML technologies and the localization process: Why a standard markup method is needed for working with multilingual documents." *MultiLingual Computing & Technology* 11 (7): 62–67.

Shei, C.-C. 2002. "Teaching machine translation and translation memory systems." Taiwan: Chang Jung University. http://www.swan.ac.uk/cals/staff/shei/publication/TMandMT.htm

Smadja, F. 1993. "Retrieving collocations from text: Xtract." *Computational Linguistics* 19 (1): 143–177.

Smadja, F. and McKeown, K. 1994. "Translating collocations for use in bilingual lexicons." *Proceedings of the ARPA Human Language Technology Workshop 94*, 152–156. Plainsboro, NJ: Advance Research Projects Agency.

Soderland, S. 1997. "Learning to extract text-based information from the World Wide Web." *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining (KDD-97)*, 251–254. Newport Beach, CA: American Association for Artificial Intelligence.

Sojitra, R. 1998. *Phrasal document analysis for modeling*. Master's Thesis, Virginia Polytechnic Institute.

Varantola, K. 2000. "Translators, dictionaries and text corpora." In *I corpora nella didattica della traduzione*, S. Bernardini and F. Zanettin (eds), 117–133. Bologna: Cooperativa Libraria Universitaria Editrice.

———. 2003. "Translators and disposable corpora." In *Corpora in Translator Education*, F. Zanettin, S. Bernardini and D. Stewart (eds), 55–70. Manchester: St. Jerome.

Webb, L.E. 1999. *Advantages and disadvantages of translation memory: A cost benefit analysis*. Master's Thesis, Monterey Institute of International Studies.

Zanettin, F. 1994. "Parallel words: Designing a bilingual database for translation activities." *Corpora in Language Education and Research: A Selection of Papers from Talc 94* [UCREL Technical Papers 4], A. Wilson and T. McEnery (eds), 99–111. Lancaster: UCREL.

———. 1998. "Bilingual comparable corpora and the training of translators." *Meta* 43 (4): 616–630.

Appendix

Localization-related standards and standards bodies

ANSI: American National Standards Institute — A private, non-profit organization that administers and coordinates the U.S. voluntary standardization and conformity assessment system (http://www.ansi.org/public/about.html). Note: ANSI is an umbrella organization that represents the diverse US national standards bodies (such as ASTM, SAE, and others) in ISO.

ANSI/ISO/ASQ. 2000. *ANSI/ISO/ASQ Q9000:2000. Quality Management Systems: Fundamentals and Vocabulary*.

———. 2000. *ANSI/ISO/ASQ Q9001:2000. Quality Management Systems: Requirements.*

———. 2000. *ANSI/ISO/ASQ Q9004:2000. Quality Management Systems: Guidelines for Performance Improvements*.

ASTM: American Society for Testing and Materials — A not-for-profit American organization that provides a global forum for the development and publication of voluntary consensus standards for materials, products, systems, and services (http://www.astm.org/cgi-bin/ SoftCart.exe/ABOUT/aboutASTM.html?L+mystore+wrls0206+1026890669). Note: ASTM's Committee F15 on Consumer Products (i.e., consumer protection) administers, among other topics, standards for negotiating agreements governing language interpretation (interpreting) and translation services.

ASTM. 2003. *ASTM F2089-01. Standard Guide for Language Interpretation Services*.

———. 2005. *ASTM F15.XXXX. Consumer-Oriented Guide to Quality Assurance in Translation*.

ATA: American Translators Association — The largest professional association of translators and interpreters in the U.S. with over 8,500 members in over 60 countries. The group is dedicated to promoting the recognition of the translation and interpretation professions, and to formulating standards for professional ethics, practices, and competence (http://www.atanet/org/). Note: The ATA Certification Committee, which certifies translators, uses the ATA Framework for Standard Error Marking as an assessment metric for grading translator certification examinations.

ATA Certification Committee. 2003. *ATA Framework for Standard Error Marking* (http://www.atanet.org/bin/view.pl/12438.html).

BSI: British Standards Institute — The national standards body of the United Kingdom (UK) that develops standards and standardization solutions and represents UK interests in the production of British, European and international standards. BSI and the US NISO group occupy leadership roles in the development of library science and knowledge management standards, which has resulted in the concept of Anglo-American standards in the area (http://www.bsi-global.com/News/Information/index.xalter). *See* ISO 2788 and ISO 5964.

BSI. BS 8723-1:2005. *Structured vocabularies for information retrieval—Guide—Definitions, symbols and abbreviations.*

———. BS 8723-2:2005. *Structured vocabularies for information retrieval—Guide—Thesauri.*

CEN: Comité Européen de Normalisation (European Committee for Standardization) — Umbrella organization founded by the national standards bodies in the European Economic Community and EFTA countries for the creation of voluntary technical standards in Europe designed to promote free trade, the safety of workers and consumers, interoperability of networks, environmental protection, exploitation of research and development programs, and public procurement.

CEN. 2006. EN 15038. *Translation services — Service requirements.*

DIN: Deutsche Institut für Normung — German Institute for Standardization, whose main activity is the development of technical rules for the benefit of the economy and of society as a whole (http://www2.din.de/index.php?lang=en). Note: DIN 2345, Translation Contracts, was one of the first national or international standards designed to provide guidance to consumers (in the broadest sense) purchasing translation services.

DIN. 1998. *DIN 2345:1998. Übersetzungsaufträge* [*Translation Contracts*]. Berlin: Beuth Verlag.

FSG: Free Standards Group — An independent, non-profit organization dedicated to accelerating the use of free and open-source software by developing and promoting standards (http://www.freestandards.org/). Note: Key projects and workgroups that fall under the Free Standards Group umbrella include the Linux Standard Base (LSB), OpenI18N (formerly Li18nux), LANANA and OpenPrinting. *See also* OpenI18n.

FSG. 2003. *The Locale Data Markup Language Specification.* Mark Davis, ed. (http://www.openi18n.org/specs/ ldml/1.0/ldml-spec.htm).

IANA: Internet Assigned Numbers Authority — An organization that assigns IP numbers and protocol parameters such as port, protocol, and enterprise numbers, as well as options, codes, and types, under the direction of the Internet Architecture Board (IAB) (http://www.atis.org/tg2k/iana.html). Note: The namespace of language tags is administered by IANA [RFC 2860] according to the rules in Section 3 of IETF 3066. Among its other Web-related activities, IANA assigns language codes for languages or sub-languages that are not included in the ISO 639 series of standards. As these standards are expanded to include all languages and eventually dialects as well, special IANA codes are being retired (http://www.ietf.org/rfc/rfc3066.txt).

IANA. 2004. *Language Tags* (http://www.iana.org/assignments/language-tags/).

ICE: Information and Content Exchange — An XML-based standard protocol for content syndication over the Internet (http://www.webopedia.com/TERM/I/Information_and_Content_Exchange.html).

ICU: International Components for Unicode — A mature, widely used set of C/C++ and Java libraries for Unicode support, software internationalization and globalization provided by IBM as free, open-source software to encourage the adoption of Unicode and standardize Unicode APIs across as many platforms as possible (http://www-306.ibm.com/software/globalization/icu/index.jsp). *See* Unicode.

IEC: International Electrotechnical Commission — The leading global organization that prepares and publishes international standards for all electrical, electronic and related technologies. *See* JTC1.

IEEE: Institute of Electrical and Electronics Engineers — A leading authority in technical areas ranging from computer engineering, biomedical technology and telecommunications, to electric power, aerospace and consumer electronics, among others, which has formulated more than 860 active standards, with another 700 under development (http://www.ieee.org/portal/index.jsp? pageID=corp_level1&path=about&file=index.xml&xsl=generic.xsl). Note: The IEEE Portable Application Standards Committee (PASC) is responsible for the POSIX standards.

IEEE. 2001. *IEEE Std 1003.1:2001 (Open Group Technical Standard, Issue 6). Standard for Information Technology — Portable Operating System Interface (POSIX®)* (http://posixcertified.ieee.org/).

IETF: Internet Engineering Task Force — A large open international community of network designers, operators, vendors, and researchers, and which is concerned with the evolution of Internet architecture and the smooth operation of the Internet (http://www.ietf.org/overview.html). Note: IETF RFC 3066 defines the parameters for xml:lang, while ISO defines the content of the language and country codes used to assemble xml:lang, and the W3C specifies that xml:lang shall be used to identify languages in the xml environment.

IETF. 1995. *IETF 1766:1995. Tags for the Identification of Languages* (http://www.ietf.org/rfc/rfc1766.txt).

———. 1998. *IETF 2413:1998. Dublin Core Metadata for Resource Discovery* (http://www.ietf.org/rfc/rfc2413.txt).

———. 2001. *IETF 3066:2001. Tags for the Identification of Languages* (http://www.ietf.org/ rfc/rfc3066.txt).

———. 2004. *IETF 3066bis:2004. Tags for Identifying Languages*, Addison Phillips and Mark Davis, author/editors (http://www.inter-locale.com/ID/draft-phillips-langtags-03.html).

IFLA: International Federation of Library Associations and Institutions — The leading international body representing the interests of library and information services and their users. Like ATA, IFLA is a professional association and not a standards body, but is currently active in generating a new model for multilingual thesauri (http://www.ifla.org/III/intro00.htm).

IFLA. 2005. *Guidelines for Multilingual Thesauri* (http://www.ifla.org/VII/s29/pubs/Draft-multilingualthesauri.pdf).

ILR: Interagency Language Roundtable — An unfunded Federal interagency organization established for the coordination and sharing of information about foreign language-related activities at the Federal level (http://www.govtilr.org/FAQs.htm). Note: The ILR is responsible for the ILR Scale, a metric designed by US government agencies for measuring language competence in US government foreign language professionals. Although this rating scale focuses on various types of language competence, it has been used in the past as a predictor of translation competence. The ILR is developing Skill Level Descriptions for Translation, which are slated at the time of publication for finalization in February, 2006.

ILR. 2004. *Interlanguage Roundtable Language Skill Level Descriptions* (http://www.govtilr.org/ILRscale1.htm, Language Proficiency Skill Level Descriptions and Scale; http://www.govtilr.org/ilrtranslationprobat_2__1_.pdf, Translation Scale).

ISA: International Federation of National Standardizing Associations — ISO predecessor organization responsible for international standardization in the mechanical engineering field between 1926 and 1942; succeeded by ISO in 1946.

ISO: International Organization for Standardization (previously ISA: International Standards Association) — A network of national standards institutes from 140 countries working in partnership with international organizations, governments, industry, business and consumer representatives, and which builds bridges between the public and private sectors (http://www.iso.ch/iso/en/ISOOnline.openerpage). ISO TC 37, *Terminology and language and content resources*, is made up of four sub-committees (SCs):

- TC 37/SC 1, Principles and methods, maintains work groups specializing in harmonization of terminology, principles, methods and vocabulary, socioterminology.
- TC 37/SC 2, Terminography and lexicography, maintains work groups specializing in language coding, terminography, lexicography, translation management, and source identification for language resources.
- TC 37/SC 3, Computer applications for terminology, maintains work groups specializing in data elements, *v*ocabulary, data interchange, database management.
- TC 37/SC 4, Language resource management, maintains work groups specializing in representation schemes, multilingual information representation, lexical resources, and workflow of language resource management.
- TC 37/TC 46 Joint Advisory Committee serves as the ISO Registration Authority (RA) for ISO 639-3.

ISO. 1988. *ISO 639-1:1988. Codes for the representation of names of languages — Part 1: Alpha-2 code*. Geneva: ISO (http://xml.coverpages.org/languageIdentifiers.html#iso639).
———. 1988. *ISO 639-2:1998. Codes for the representation of names of languages — Part 2: Alpha-3 code.* Geneva: ISO (http://lcweb.loc.gov/standards/iso639-2/langcodes.html).
———. 2003. *ISO CD 639-3:2003. Code for the representation of names of languages — Part 3: Alpha-3 code for comprehensive coverage of languages.* Geneva: ISO.
———. 2003. *ISO NWI 639-4:2003. Code for the representation of names of languages — Part 4: Implementation guidelines and general principles for language coding*. Geneva: ISO.
———. 2003. *ISO NWI 639-5:2003. Code for the representation of names of languages — Part 5: Alpha-3 code for language families and groups*. Geneva: ISO.
———. 2003. *ISO NWI 639-6:2003. Code for the representation of names of languages — Part 6: Alpha-4 code for dialects*. Geneva: ISO.
———. 2000. *ISO 704:2000. Terminology work — Principles and methods.* Geneva: ISO.
———. 2000. *ISO 1087-1:2000. Terminology work — Vocabulary — Part 1: Theory and application*. Geneva: ISO.

———. 2000. *ISO 1087-2:2000. Terminology work — Vocabulary — Part 2: Computer applications.* Geneva: ISO.
———. 1996. *ISO 2709:1996. Information and documentation — Format for Information Exchange.* Geneva: ISO.
———. 1986. *ISO 2788-1986. Guidelines for the establishment and development of monolingual thesauri.* Geneva: ISO. *See* BS 5723:1987.
———. 1997. *ISO 3166:1997. Codes for the representation of names of countries and their subdivisions — Part 1: Country codes.* Geneva: ISO (http://ftp.ics.uci.edu/pub/ietf/http/related/iso3166.txt; http://www.unicode.org/unicode/onlinedat/countries.htm).
———. 1985. *ISO 5964-1985. Guidelines for the establishment and development of multilingual thesauri.* Geneva: ISO. *See* BS 6723:1985.
———. 1998–2001. *ISO/IEC 8859-1:1998–2001. Information technology — 8-bit single-byte coded graphic character set — Parts 1–16.* Geneva: ISO (http://www.iso.ch; http://czyborra.com/charsets/iso8859.html).
———. 2005. *IS0 9000. Quality management systems—Fundamentals and vocabulary*. Geneva: ISO (http://www.iso.ch/iso/en/CatalogueListPage.CatalogueList?COMMID=4166&scopelist=).
———. 2000. *ISO 10646-1:2000. Information technology — Universal multiple-octet coded character set (UCS) — Part 1: Architecture and basic multilingual plane*. Geneva: ISO (http://anubis.dkuug.dk/JTC1/SC2/WG2/docs/standards).
———. 2000. *ISO 12199:2000(E). Alphabetical ordering of multilingual terminological and lexicographical data represented in the Latin alphabet.* Geneva: ISO (http://www.iso.ch/iso/en/CombinedQueryResult.CombinedQueryResult?queryString=ISO+12199%3A2000).
———. 1999. *ISO 12620:1999. Computer applications in terminology — Data categories.* Geneva: ISO.
———. 2004. *ISO 15924:2004. Information and documentation — Codes for the representation of names of scripts.* Geneva: ISO.
———. 2003. *ISO 16642:2003. Computer applications in terminology — TMF (Terminological Markup Framework).* Geneva: ISO.
———. 1998. *ISO 23950:1998. Information and documentation — Information retrieval* . Geneva, ISO. *See* NISO Z39.50 — *Application service definition and protocol specification.*
———. 2004. *ISO WD 24613:2004. Lexical resource markup framework (LMF).* Geneva: ISO.

JTC1: ISO/IEC Joint Technical Committee 1 — Joint technical committee of ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) whose mission is to develop, maintain, promote and facilitate IT standards required by global markets (http://www.jtc1.org/Navigation.asp?Mode=Browse&Area=Glance&SubComm=ISO%2FIECJTC1&CommLevel=TC&OldSubComm=ISO%2FIEC+JTC+1&SCCODE=). Note: Historically, the role of JTC1 evolved out of the growing presence of electronic (computerized) applications in virtually all areas of science and industry, which led on occasion to conflicts between ISO and IEC technical committees. JTC1's relationship to language standards is particularly evident with respect to its administration of standards for characters sets (SC2, Coded character sets), Standard Generalized Markup Language, SGML (SC 34, Document Description and Processing Languages) and data elements and metadata registries (SC 32, Data management and Interchange). *See also* the Home Page for *ISO/IEC 11179 Information Technology — Metadata Registries* (http://metadata-stds.org/11179/).

JTC1. 1986. *ISO/IEC 8879:1986. Information processing — Text and office systems — Standard Generalized Markup Language (SGML)*. Geneva: ISO.
———. 2003. *ISO/IEC 10646:2003. Universal multiple-octet coded character set (UCS)*. Geneva: ISO.
———. 2004. *ISO/IEC 11179-1:2004. Information technology—Metadata registries (MDR)—Part 1: Framework*. 2nd edition. Geneva: ISO.
———. 2005. *ISO/IEC 11179-2:2005. Information technology—Metadata registries (MDR)—Part 2: Classification*. 2nd edition. Geneva: ISO.
———. 2003. *ISO/IEC 11179-3:2003. Information technology—Metadata registries (MDR)—Part 3: Registry metamodel and basic attributes*. Geneva: ISO.
———. 2004. *ISO/IEC 11179-4:2004. Information technology—Metadata registries (MDR)—Part 4: Formulation of data definitions*. 2nd edition. Geneva: ISO.
———. 2005. *ISO/IEC 11179-5:2005. Information technology—Metadata registries (MDR)—Part 5: Naming and identification principles*. 2nd edition. Geneva: ISO.
———. 2005. *ISO/IEC 11179-6:2005. Information technology—Metadata registries (MDR)—Part 6: Registration*. 2nd edition. Geneva: ISO.
———. 1999. *ISO/IEC 13250. Information technology — Document description and processing languages — Topic Maps.* Geneva: ISO.

LISA: Localization Industry Standards Association — Professional organization consisting of over 400 leading IT manufacturers and services providers dedicated to providing best practice, business guidelines and multi-lingual information management standards for making enterprise globalization become a reality (http://www.lisa.org/). *See also* OSCAR.

LISA. 2004. *LISA GILT Metrics eXchange Specification* [Sections: GMX-V = GILT Metrics-Volume; GMX-C = GILT Metrics-Complexity; GMX-Q = GILT Metrics-Quality] (http://www.lisa.org/oscar/gmx/).
———. 2004. *LISA Global Content Management Guide* (http://www.lisa.org/interact/gcms.html).
———. 2005. *LISA QA Model 3.0: License Agreement and Product Documentation*. Fechy, Switzerland: LISA (http://www.lisa.org/products/qamodel.html).
———. *LISA Outsourcing Guidelines* (http://www.lisa.org/tmp/Localization_Project_Outsourcing_Guidelines.pdf).
———. *LISA Segmentation Rules eXchange (SRX)* (http://www.lisa.org/oscar/seg/).
———. *LISA TermBase eXchange (TBX)* (http://www.lisa.org/tbx/).
———. *LISA Translation Memory eXchange (TMX)* (http://www.lisa.org/tmx/).
———. 2001. *Localizing for Mobile Devices: A Primer* (http://www.lisa.org/interact/wireless/).

Localization Institute — An organization providing quality resources, training, seminars, and conferences on localization, internationalization, and international business development (http://www.localizationinstitute.com/). Note: The L10n Institute is not affiliated with any localization service or tool provider. The Institute has launched an industry-wide metric project designed to create quantifiable business and productivity metrics (http://localizationinstitute.com/surveysite/).

MARTIF: Machine-Readable Terminology Interchange Format. *See* LISA: TBX.

NISO: National Information Standards Association (US) — A non-profit association accredited by the American National Standards Institute (ANSI) that identifies, develops, maintains, and publishes technical standards to manage information in traditional and digital environments. NISO is responsible for developing a number of international standards, such as ISO 639-2 and ISO 23950. NISO is the ANSI Technical Advisory Group (TAG) for ISO Technical Committee 46 on Information and Documentation and functions as ANSI TC Z39 (http://www.niso.org/about/index.html).

NISO. 2005. *NISO Z39.19:2005. Guidelines for the construction, format, and management of monolingual controlled vocabularies* (http://www.niso.org/standards/resources/z39-19-2005.pdf).

OASIS: Organization for the Advancement of Structured Information Standards — A not-for-profit, global consortium that drives the development, convergence and adoption of e-business standards (http://www.oasis-open.org/who/). Note: Among other things, OASIS is responsible for the XLIFF standard, whose purpose is to store localizable data and carry it from one step of the localization process to the other, while allowing interoperability between tools, and for DITA, which "is an XML-based, end-to-end architecture for authoring, producing, and delivering technical information. This architecture consists of a set of design principles for creating 'information-typed' modules at a topic level and for using that content in delivery modes such as online help and product support portals on the Web" (http://www.oasis-open.org/committees/xliff/documents/xliff-specification.htm).

OASIS. 2003. *XLIFF 1.1 Specification [XML Localization Interchange File Format]* (http://www.oasis-open.org/committees/xliff/documents/xliff-specification.htm).

———. 2005. *DITA, Darwin Information Typing Architecture* (http://www-128.ibm.com/developerworks/xml/library/x-dita1/index.html).

———. 2005. *Translation Web Services Specification. Draft Specification* (http://www.oasis-open.org/committees/download.php/13158/trans-ws-spec.html).

OLIF2: Open Lexicon Interchange Format — XML-compliant interchange format for translation-related lexical and/or terminological entries intended for interchange between different types of lexicons, particularly involving MT lexicons (http://www.olif.net/).

OMG: The Object Modeling Group is responsible for defining and maintaining the Unified Modeling Language (UML), a four part standard that is used for modeling metamodels, concept systems, business models, and a variety of other applications. The standard is widely used and supported by tools and Web-based tutorials.

OMG. Unified Modeling Language (UML) Specification: Superstructure version 2.0 (http://www.uml.org/#UML2.0).

———. Unified Modeling Language (UML) Specification: Infrastructure version 2.0 (http://www.omg.org/docs/ptc/04-10-14.pdf).

———. Unified Modeling Language (UML) Object Constraint Language (OCL) 2.0 Specification (http://www.omg.org/docs/ptc/05-06-06.pdf).

———. Unified Modeling Language (UML): Diagram Interchange version 2.0 (http://www.omg.org/docs/ptc/05-06-04.pdf).

ON: Österreichische Normungsinstitut — Austrian Standards Institute (http://www.oenorm.at/). Note: The ON is responsible for the ON D 1200 and D 1201 standards governing translation and interpretation services.

ON. *ON D 1200:2000. Translation and interpretation services, Translation services. Requirements of the service and the provision of the service*.

———. *ON D 1201:2000. Translation and interpretation services, Translation services. Translation contracts*.

OpenI18n: Free Standards Group Open Internationalization Initiative — The Free Standards Group task force responsible for the locale data markup language specification version 1.0, which is the foundation of the Common XML Locale Repository project (http://www.openi18n.org/modules.php?op=modload&name=O-News&file=article&sid=6&mode=thread&order=0&thold=0). Note: The purpose of this project is to devise a general XML format for the exchange of culturally sensitive (locale) information for use in application and system development, and to gather, store, and make available data generated in that format (see http://www.openi18n.org/specs/ldml/). This Linux-based standard is closely associated with the development of IETF 3066bis, which specifies locale notation for use in Internet environments (http://www.openi18n.org/modules.php?op=modload&name=O-News&file=article&sid=6&mode=thread&order=0&thold=0). *See* FSG.

OpenTag Format — A standard markup format for storing, manipulating, and using extracted text for localization purposes. Note: OpenTag reflected the evolution to XML of an original extraction format developed at ILE in Colorado in the late 80s and early 90s. The concept behind the format was taken up by the XLIFF group and became an OASIS Committee Specification in 2002. (http://www.opentag.com/xliff.htm#DiffWithOTF).%20See%20XLIFF). *See* XLIFF.

OSCAR: Open Standards for Container/Content Allowing Re-use — The LISA Special Interest Group (SIG) responsible for the definition of such standards as TMX (Translation Memory eXchange format) and TBX (TermBase eXchange format) (http://www.ttt.org/oscar/index.htm). *See* LISA.

POSIX: Portable Operating System Interface — A set of programming interface standards governing how to write application source code so that the applications are portable between operating systems. The POSIX standards include specifications for locale references. *See* IEEE.

SAE: Society of Automotive Engineers — The American standards organization responsible for standards used in designing, building, maintaining, and operating self-propelled vehicles for use on land or sea, in air or space (http://www.sae.org/about/index.htm). Note: SAE is responsible for the administration of the *SAE J2450 Translation Quality Metric*.

SAE. 2001. *SAE J2450:2001. Translation Quality Metric*.

SGML: Standard Generalized Markup Language. *See also* JTC1, SGML.

SIL: SIL International, formerly Summer Institute for Linguistics — A non-profit, scientific educational organization of Christian volunteers that specializes in serving the lesser-known language communities of the world and in the application of linguistic research to the literacy and translation needs of minority language communities (http://www.sil.org/sil/faq.htm). Note:

SIL International is responsible for creating Ethnologue.com, a listing of the world's 6000-plus languages. This collection forms the basis of ISO 639-3 (http://www.ethnologue.com/web.asp).

SIL International. 2005. Ethnologue: Languages of the World, 15th ed., Raymond G. Gordon, ed. Dallas, TX: SIL InternationalSRX: Segmentation Rules eXchange. *See* LISA: SRX.

TBX: TermBase eXchange. *See* LISA: TBX.

TEI: Text Encoding Initiative — An international research effort established in 1987, intended to produce a community-based standard for encoding and interchange of texts. The work began in SGML and is in the process of migrating to XML (http://www.tei-c.org/Consortium/TEIcharter.html). Note: TEI Work Group 13 developed the TIF (Terminology Interchange Format) as a component of the original TEI project. TIF and its relatively short-lived ETIF incarnation evolved into ISO 12200 and eventually into LISA/OSCAR's TBX format. Representatives of TEI are currently involved in the work of ISO TC 37/SC 4.

TMX: Translation Memory eXchange. *See* LISA, TMX.

TopicMaps.org. — An independent consortium of parties interested in developing the applicability of the Topic Maps Paradigm to the World Wide Web, by leveraging the XML family of specifications as required (http://www.topicmaps.org/#welcome). TopicMaps.org is responsible for the XML implementation of ISO 13250, the topic maps standard.

TopicMaps.org. 2001. *XML Topic Maps 1.0. (XTM): TopicMaps.org Specification* (http://www.topicmaps.org/xtm/1.0/).

UCS: Universal Coded Character Set. *See* JTC1 and Unicode Consortium.

Unicode Consortium — A non-profit organization originally founded to develop, extend and promote use of the Unicode Standard, which specifies the representation of text in modern software products and standards (http://www.unicode.org/consortium/consort.html). Note: The Unicode Standard is a character coding system designed to support the worldwide interchange, processing, and display of the written texts of the diverse languages and technical disciplines of the modern world. In addition, it supports classical and historical texts of many written languages. The name is derived from three main goals, in that it is (a) universal (addresses the needs of world languages); (b) uniform (uses fixed-width codes for efficient access); and (c) unique (each bit sequence has only one interpretation in character code). Unicode is synonymous with ISO 10646 and has as its mission the representation of all significant human languages (http://www.unicode.org/history/summary.html).

Unicode Consortium. 2003. *The Unicode Standard, Version 4.0.* Boston, MA, Addison-Wesley.
———. Unicode Standard Annex # 24 (USA#24). *Script Names* (http://www.unicode.org/reports/tr24/). (This document assigns script names from ISO 15942 to all Unicode code points.)
———. 2004. Unicode Technical Standard # 35 (UTS#5). *Locale Data Markup Language (LDML)* (http://www.unicode.org/reports/tr35/).
———. 2005. Unicode Character Database (UCD) (http://www.unicode.org/ucd/).

W3C: World Wide Web Consortium — The organization responsible for developing common protocols that promote the evolution of the World Wide Web. These standards include interoperable technologies in the form of specifications, guidelines, software, and tools. W3C is dedicated to future development of the Web as a robust, scalable, and adaptive infrastructure for world-wide information access and interchange (http://www.w3.org/). *See also* W3C I18n.

W3C. 2004. *Extensible Markup Language (XML)* (http://www.w3.org/XML/).
W3C Internationalization (I18n): World Wide Web Consortium Internationalization Activity — Task force within the W3C whose goal is to propose and coordinate techniques, conventions, guidelines and activities within the W3C and together with other organizations that facilitate the use of W3C technology worldwide, with different languages, scripts, and cultures (http://www.w3.org/ International/).
W3C I18n. 2003. *Editing Internationalized Resource Identifiers (IRIs)* (http://www.w3.org/International/iri-edit/).
———. 2003. *Requirements for the Internationalization of Web Services* (http://www.w3.org/TR/ws-i18n-req/).
———. 2004. *Authoring Techniques for XHTML & HTML Internationalization: Characters and Encodings 1.0* (http://www.w3.org/TR/i18n-html-tech-char/).
———. 2004. *Authoring Techniques for XHTML & HTML Internationalization: Specifying the Language of Content 1.0* (http://www.w3.org/TR/i18n-html-tech-lang/).
———. 2004. *Authoring Techniques for XHTML & HTML Internationalization: Handling Bidirectional Text 1.0* (http://www.w3.org/TR/i18n-html-tech-bidi/).
———. 2004. *The Character Model for the World Wide Web 1.0: Fundamentals (CHARMOD)* (http://www.w3.org/TR/charmod/).
———. 2004. *OWL Web Ontology Language Overview: W3C Recommendation 10 February 2004* (http://www.w3.org/TR/owl-features/).
———. 2005. *SKOS Core Guide [Simple Knowledge Organization System]: W3C Working Draft 10 May 2005* (http://www.w3.org/TR/2005/WD-swbp-skos-core-guide-20050510/).
———. 2006. *Internationalization Tag Set (ITS), W3C Working Draft 22 November 2005* (http://www.w3.org/International/its/itstagset/itstagset.html).

XML: eXtensible Markup Language — *See* W3C, XML.

XLIFF: XML Localization Interchange File Format — *See* OASIS: XLIFF.

Suggestions for further reading

BabelPort (Language industry news and resources). http://www.babelport.com/

Bishop, M. 1998. *How to Build a Successful International Web Site: Designing Web Pages for Multilingual Markets at the National and International Level*. New York: Coriolis Group Books.

Bowker, L. 2002. *Computer-Aided Translation: A Practical Introduction*. Ottawa: University of Ottawa Press.

Chandler, H.M. 2004. *The Game Localization Handbook*. Hingham, MA: Charles River Media.

Common Sense Advisory, Inc. *The Global Watchtower. Translation, Localization, Globalization, Internationalization Industry News*. http://www.commonsenseadvisory.com/en/news/global_watchtower.php

Coulmas, F. 1999. *The Blackwell Encyclopedia of Writing Systems*. Cambridge, MA, USA/Oxford, UK: Blackwell Publishers.

Daniels, P.T. and Bright, W. (eds). 1996. *The World's Writing Systems*. New York: Oxford UP.

Deitsch, A. and Czarnecki, D. 2001. *Java Internationalization.* Beijing/Sebastopol, CA: O'Reilly.

del Galdo, E.M. and Nielsen, J. (eds). 1996. *International User Interfaces.* New York: Wiley.

DePalma, D.A. 2002. *Business without Borders: A Strategic Guide to Global Marketing.* New York: Wiley.

Dowling, P. 1998. "Localizing for lands beyond the wild frontier." *Gamasutra*, August 28. Originally published in the May 1998 issue of *Game Developer*. http://gamasutra.com/features/production/19980828/localization_01.htm

Dr. International (ed). 2002. *Developing International Software.* 2nd ed. Redmond, WA: Microsoft.

Ebben, S. and Marshall, G. 1999. "Localization process: Globalizing your code and localizing your site." *Microsoft TechNet*, March 21. http://www.microsoft.com/technet/archive/ittasks/plan/sysplan/glolocal.mspx

Esselink, B. 2000. *A Practical Guide to Localization.* Amsterdam/Philadelphia: John Benjamins.

Gamasutra. The Art & Science of Making Games. http://www.gamasutra.com/

Gillam, R. 2003. *Unicode Demystified: A Practical Programmer's Guide to the Encoding Standard*. Boston, MA: Addison-Wesley.

Goodpasture, J.C. 2004. *Quantitative Methods in Project Management*. Boca Raton, FL: J. Ross.

Gordon, R.G., Jr. (ed). 2005. *Ethnologue: Languages of the World*. 15th ed. Dallas, TX: SIL International. Online version: http://www.ethnologue.com/

Graham, T. 2000. *Unicode: A Primer*. Foster City, CA: M&T Books.

Hoft, N.L. 1995. *International Technical Communication: How to Export Information about High Technology.* New York: Wiley.

Ibbs, W. and Reginato, J. 2002. *Quantifying the Value of Project Management: Best Practices for Improving Project Management Processes, Systems, and Competencies*. Newtown Square, PA: Project Management Institute.

Lientz, B.P. and Rea, K.P. 2003. *International Project Management*. Amsterdam/Boston: Academic Press.

Lunde, K. 1999. *CJKV Information Processing.* Beijing/Cambridge: O'Reilly.

Microsoft Corporation. Microsoft glossaries of translated user interface terms. ftp://ftp.microsoft.com/developr/msdn/newup/glossary

Microsoft Corporation Editorial Style Board. 2004. *Microsoft Manual of Style for Technical Publications*. 3rd ed. Redmond, WA: Microsoft Press.

Microsoft Global Development and Computing Portal. 2002. "Globalization step-by-step: Introduction." http://www.microsoft.com/globaldev/getwr/steps/wrguide.mspx

———. 2005. "Designing a world-ready program." http://www.microsoft.com/globaldev/getWR/DesignWRProg/default.mspx

Miozzo, M. and Miles, I. (eds). 2002. *Internationalization, Technology, and Services*. Cheltenham, UK/Northhampton, MA, USA: Edward Elgar.

Nakanishi, A. 1990. *Writing Systems of the World: Alphabets, Syllabaries, Pictograms*. Rev. ed. Rutland, VT/Tokyo: C.E. Tuttle Co.

Nielsen, J. 1993. *Usability Engineering*. Boston: Academic Press.

———. 2000. *Designing Web Usability: The Practice of Simplicity*. Indianapolis, IN: New Riders.

Nielsen, J., et al. 2001. *E-Commerce User Experience*. Fremont, CA: Nielsen Norman Group.

Ott, C. 1999. *Global Solutions for Multilingual Applications: Real-World Techniques for Developers and Designers.* New York: Wiley.

Owen, R.K. 2000. "Yen, marks, kroner and pesetas: Squeezing every last penny out of international markets." *Gamasutra*, January 10. Originally published in the *Computer Game Developers Conference 1998 Proceedings*. http://www.gamasutra.com/features/20000110/owen_01.htm

Project Management Institute. 2002. *The PMI Compendium of Project Management Practices*. Newtown Square, PA: Project Management Institute.

———. 2004. *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*. 3rd ed. Newtown Square, PA: Project Management Institute.

Rose, K.H. 2005. *Project Quality Management: Why, What, and How*. Boca Raton, FL: J. Ross.

Savourel, Y. 2001. *XML Internationalization and Localization.* Indianapolis, IN: SAMS.

Schmidt, M. 2000. *Implementing the IEEE Software Engineering Standards*. Indianapolis, IN: SAMS.

Singh, N. 2005. *The Culturally Customized Web Site*. Boston: Elsevier.

Smith, P.G. 2002. *Proactive Risk Management: Controlling Uncertainty in Product Development.* New York: Productivity Press.

Sprung, R.C. (ed). 2000. *Translating into Success. Strategies for Going Multilingual in a Global Age*. Amsterdam/Philadelphia: John Benjamins.

Sun Technical Publications. 1996. *Read Me First! A Style Guide for the Computer Industry*. [Upper Saddle River, NJ]: Prentice Hall.

Symmonds, N. 2002. *Internationalization and Localization Using Microsoft .Net.* Berkeley, CA: APress.

Taylor, D. 1992. *Global Software: Developing Applications for the International Market*. New York: Springer-Verlag.

Trompenaars, F. and C. Hampden-Turner. 1998. *Riding the Waves of Culture: Understanding Cultural Diversity in Business*. 2nd Edition. New York: McGraw-Hill.
Tuthill, B. and D. Smallberg. 1997. *Creating Worldwide Software: Solaris International Developer's Guide*. 2nd edition. Mountain View, CA: Sun Microsystems Press.
Verdin, P. and Van Heck, N. 2001. *From Local Champions to Global Masters: A Strategic Perspective on Managing Internationalization.* New York: Palgrave Macmillan.
Wright, S.E. and Budin, G. 2001. *Handbook of Terminology Management*. Amsterdam/Philadelphia: John Benjamins.
Wysocki, R.K., with McGary, R. 2003. *Effective Project Management: Traditional, Adaptive, Extreme*. 3rd ed. Indianapolis, IN: Wiley.
Yunker, J. 2002. *Beyond Borders: Web Globalization Strategies.* Indianapolis, IN: New Riders.
Zervaki, T. 2002. *Globalize, Localize, Translate: Tips and Resources for Success*. London: Authorhouse.

Contributors

Scott Bass is the founder and president of Advanced Language Translation Inc., an ISO 9001:2000-certified provider of translation and localization services. He is a member of the American Translators Association, the Society for Technical Communications, and the World Trade Center Buffalo/Niagara, and serves on the Advisory Board of the International Business Council of the Rochester Business Alliance. He has lived, studied and worked in both Austria and Germany.

Don DePalma is the founder and president of Common Sense Advisory, Inc, a research and consulting firm specializing in the on- and offline operations driving business globalization, internationalization, translation, and localization. He is the author of *Business Without Borders: A Strategic Guide to Global Marketing* (2002), co-chair of the Internationalization and Unicode Conference committee, and a member emeritus of the Board of Directors of the Globalization and Localization Association. Previously Don was the vice president of corporate strategy at Idiom Technologies and a principal analyst at Forrester Research.

Frank Dietz, Ph.D. is a native of Germany and an ATA-certified English-German translator who lives in Austin, Texas. He specializes in technical translation and software localization and has translated over 30 computer games into German, among them *System Shock* and *System Shock 2*, *Wing Commander 3* and many of the titles in the *Jane's Combat Simulation* series. His Web site www.frankdietz.com contains a glossary of game localization terminology.

Carla DiFranco is a Program Manager at Microsoft, specializing in translation tools, recycling strategies and workflow improvement. Previously she worked as a freelance translator (German to English patent translation) as well as operations manager of a localization company, specializing in Web-based training courses. She has been teaching terminology, translation and computer applications for translators since 1997.

Susan M. Dray holds a doctorate in Psychology from UCLA and is a Board-Certified Human Factors Professional. As President of Dray & Associates, Inc., she consults internationally on interface design and usability and specializes in international user studies, including ethnographic and contextual field research and usability studies. Her user research includes one of the largest international usability evaluations ever conducted, involving 120 users in 8 countries in Europe and Asia. A leader in the Human Factors profession nationally and internationally, she has given over 100 presentations at conferences and symposia in the U.S., Europe, Africa, Australia, and New Zealand, and has published numerous papers and book chapters. For many years, she co-edited (with David A. Siegel) the Business Column of the Association for Computing Machinery's magazine *interactions*.

Keiran J. Dunne is an Assistant Professor of French and a member of the faculty in the Institute for Applied Linguistics at Kent State University, where he teaches graduate courses in localization as well as undergraduate and graduate courses in French civilization and advanced composition. Drawing upon nearly a decade's experience as a localization and technical translation subcontractor for Fortune 500 companies and other corporate clients, his research interests include localization and project management. He is currently working on a computer-assisted translation textbook.

Debbie Folaron is an Assistant Professor in Translation at Concordia University, Montreal, Quebec, where she teaches specialized translation, translation technologies, contemporary translation theories, and localization. Her research includes translation and new technology laws, applied social and communications network theories to translation and localization, and online teaching. Previously, Debbie was Language and Technology Manager at Eriksen Translations Inc. in New York, and an in-house translator and Telecommunications Manager for import-export and shipping companies in Spain.

Eric Heimburg led the localization technical development for "Asheron's Call 2," an MMORPG from Turbine Entertainment. He currently works as a systems designer on Perpetual Entertainment's upcoming MMORPG, "Star Trek Online."

Barbara Inge Karsch holds a Bachelor's equivalent (Sprachen- und Dolmetscher-Institut, Munich) and a Master's Degree (Monterey Institute of International Studies, Monterey, CA) in translation and interpretation for German and English. At J.D. Edwards she drove design and implementation of a terminology management system for over 80 users. More recently, she has been working on getting the new TMS at Microsoft off the ground.

Robin Lombard joined Microsoft in 1999 and became the U.S. terminologist when the role was created in 2001. She is responsible for driving standard U.S. English terminology across Microsoft. This includes working with employees to understand terminology issues from the perspective of both content publishers and localizers, helping establish tools and processes for managing terminology, and creating educational programs. Robin holds M.A. and Ph.D. degrees in Linguistics as well as a B.S. in Science.

Arle Lommel has worked for the Localization Industry Standards Association (LISA) since 1997, where he is currently Publications Manager and Survey Analyst. His academic background is in linguistics and semiotics, and he is currently working on a doctoral degree in Folklore Studies at Indiana University. He can be reached at alommel@indiana.edu.

Clove Lynch is a localization technology strategist, implementation specialist and designer. He holds an MA in Translation from Kent State University and is an ATA-certified translator. He has held positions as Translation Tools Manager at International Language Engineering Corporation, R&D Support and Implementation Manager at International Communications and Principal Product Manager at GlobalSight Corporation. He has published articles and essays on terminology management and the ROI of Translation Memory, and occasionally speaks at localization conferences.

Gregory M. Shreve is Chairman of the Department of Modern and Classical Language Studies and Founder/Director of the Institute for Applied Linguistics at Kent State University, the United States' foremost university-based translator training program. Shreve is the general editor of the monograph series *Translation Studies* and co-author of several influential books and anthologies on translation studies including *Translation as Text* with Albrecht Neubert and *Cognitive Processes in Translation and Interpreting* with Joseph H. Danks. Shreve has broad research and teaching interests in translation theory and practice, computer-assisted translation, corpus linguistics, internationalization, software localization and language informatics.

David A. Siegel holds a doctorate in psychology from UCLA, and has worked with Dray & Associates since 1993. Through field user studies and contextual research, formal usability evaluation, and expert evaluation of interface designs, he helps development teams create useful and usable products that fit their usage contexts. He has consulted on many software applications, Web designs, and designs for new technologies. David has published articles and book chapters and taught on a variety of user-centered design topics, including recent workshops and tutorials at professional conferences in the U.S., Europe, and Africa. He has been the co-editor (with Susan M. Dray) of the Business Column of the Association for Computing Machinery's magazine, *interactions*.

Sue Ellen Wright is chair of the ATA Terminology Committee and is an ATA-accredited translator (German-English). She is active in the national and international standards community (American Society for Testing and Materials and the International Organization for Standardization) and chairs the U.S. mirror committee (Technical Advisory Group) for ISO Technical Committee 37, *Terminology and language and content resources*. She is a Professor of German and a member of the faculty in the Institute for Applied Linguistics at Kent State University, where she teaches computer applications for translators and German to English technical translation. She is the co-compiler (with Professor Gerhard Budin of the University of Vienna) of the *Handbook of Terminology Management* and the author of many articles on applied terminology management in industry. She is active as a terminology trainer.

Index

Page numbers in italics refer to figures and tables.

S

T

In the series *American Translators Association Scholarly Monograph Series* the following titles have been published thus far or are scheduled for publication:

XIII **DUNNE, Keiran J. (ed.):** Perspectives on Localization. 2006. vi, 356 pp.

XII **BAER, Brian James and Geoffrey S. KOBY (eds.):** Beyond the Ivory Tower. Rethinking translation pedagogy. 2003. xvi, 259 pp.

XI **SPRUNG, Robert C. (ed.):** Translating Into Success. Cutting-edge strategies for going multilingual in a global age. 2000. xxii, 240 pp.

X **FISCHBACH, Henry (ed.):** Translation and Medicine. 1998. viii, 180 pp.

IX **LABRUM, Marian B. (ed.):** The Changing Scene in World Languages. Issues and challenges. 1997. 160 pp.

VIII **MORRIS, Marshall (ed.):** Translation and the Law. 1995. viii, 334 pp.

VII **HAMMOND, Deanna L. (ed.):** Professional Issues for Translators and Interpreters. 1994. viii, 216 pp.

VI **WRIGHT, Sue Ellen and Leland D. WRIGHT, JR. (eds.):** Scientific and Technical Translation. 1993. viii, 298 pp.

V **LARSON, Mildred L. (ed.):** Translation: Theory and Practice, Tension and Interdependence. viii, 270 pp. *Expected Out of print*

IV **BOWEN, David and Margareta BOWEN (eds.):** Interpreting. Yesterday, Today, and Tomorrow. viii, 184 pp. *Expected Out of print*

III **KRAWUTSCHKE, Peter W. (ed.):** Translator and Interpreter Training and Foreign Language Pedagogy. vi, 178 pp. *Expected Out of print*

II **VASCONCELLOS, Muriel (ed.):** Technology as Translation Strategy. viii, 248 pp. *Expected Out of print*

I **GADDIS ROSE, Marilyn (ed.):** Translation Excellence. *Expected Out of print*