# Lokalisierung einer Entwicklungsumgebung mit einem Nicht-Standard-System am Beispiel von *L10N*

# Diplomarbeit

zur Erlangung des akademischen Grades

# Diplom-Fachübersetzerin (Fachhochschule)

eingereicht an der

# Hochschule Anhalt (FH) Hochschule für angewandte Wissenschaften Fachbereich Informatik

Betreuer (HS Anhalt (FH)): Betreuer (Georg Heeg eK):

Ausgabetermin: Abgabetermin:

Prof. Dr. Uta Seewald-Heeg Dipl.-Inf. Georg Heeg

20.07.2005 06.09.2005

# Inhaltsverzeichnis

1 Einleitung	4
2 Grundlagen	5
2.1 Lokalisierung Internationalisierung Globalisierung	5
2.7 Übersetzungswissenschaftliche Grundlagen	
3 Internationalisierung	
3.1 Gründe für die Internationalisierung von Software- produkten	
3.2 Folgen unzureichender Internationalisierung	8
3.3 Merkmale eines gut internationalisierten Software-produktes	
3.3.1 Allgemeines	
3.3.2 Stil	
3.3.3 Textexpansion	14
3.4 Kontrollierte Sprache	14
3.5 Externalisierung der übersetzbaren Daten	
3.6 Internationalisierung von VisualWorks	
3.6.1 Allgemeines	
3.6.2 Funktionsweise von Nachrichtenkatalogen	
3.6.3 Mehrere Kataloge pro Locale verwenden	17
3.6.4 Die Suche mit mehreren Katalogen optimieren	
3.6.5 Fehlgeschlagene Suche	
4 Computergestützte Ubersetzung	19
4.1 Allgemeines	
4.2 Translation-Memory-Systeme	
4.2.1 Allgemeines	
4.2.2 Der Übersetzungseditor	
4.2.3 Die Terminologieverwaltung	
4.2.4 Das Translation Memory	
4.2.5 Maschinelle Übersetzung mit Translation-Memory-Systemen	
4.2.6 Statistiken und Berichte	
4.2.7 Unterstützte Formate	
4.3 Lokalisierungstools	
4.3.1 Allgemeines	
4.3.2 Passolo	
4.3.2.1 Allgemeines	
4.3.2.2 Grundlegende Aspekte bei der Übersetzung mit Passolo	
4.3.2.3 Weitere Leistungsmerkmale von Passolo	
4.3.2.4 Schnittstellen für Glossare und Translation-Memory-Systeme	
4.3.2.5 Von Passolo unterstutzte Datenormate	
4.3.3 Catalyst	
4.3.3.1 Aligementes.	
4.3.3.2 Grundlegende Aspekie der Übersetzung mit Catalyst	
4.3.3.5 Schillustenen für Glossare und Fransfation-Memory-Systeme	
4.5.5.4 von Cataryst unterstutzte Datenormate	

5 Das Lokalisierungstool L10N	
5.1 Besonderheiten von VisualWorks	
5.2 Allgemeines zu LION	
5.3 Elemente der Benutzeroberfläche	
5.4 Evaluierung	
5.4.1 Allgemeines	
5.4.2 Stärken des Tools	62
5.4.3 Verbesserungspotenzial	
5.4.4 Zusammenfassung der Evaluierung	. 67
6 Lokalisierung von VisualWorks 7.3.1 - übersetzungswissenschaftliche Asnekte	68
6.1 Allgemeines	68
6.2 Die Übersetzung von EDV-Texten	68
6.3 Stil deutscher und englischer EDV-Texte	69
6.4 Schwierigkeiten bei der Übersetzung von EDV-Texten	
6.5 Übersetzungsstrategien	
7 Fazit und Perspektiven	79
Literaturverzeichnis	80
Anhang	84
ThelocalizationtoolLlON	84
1 Introduction	84
2 The user interface	85
3 Evaluation	92
3.1 Introduction	92
3.2 Advantages	92
3.3 Suggestions for improvement	
3.4 Summary.	96
Eidesstattliche Erklärung	97

# 1 Einleitung

Die vorliegende Diplomarbeit befasst sich mit der Lokalisierung der Menü-Elemente der Benutzeroberfläche der Smalltalk-Entwicklungsumgebung *VisualWorks 7.3.1* des Herstellers *Cincom Systems, Inc.* vom Englischen ins Deutsche mit dem Lokalisierungswerkzeug *L10N* der Firma *Georg Heeg eK*, die die Autorin im Rahmen dieser Arbeit durchführte. Als Zielgruppe der Lokalisierung wird ein Personenkreis angenommen, der das Programmieren mit Smalltalk erlernen möchte, aber nur begrenzte Englischkenntnisse aufweist.

Ausgehend von der sprachlichen Anpassung der Benutzeroberfläche werden übersetzungswissenschaftliche Aspekte der Lokalisierung sowie die Internationalisierung einer Software als Voraussetzung für eine qualitativ hochwertige Lokalisierung dieser Software im Allgemeinen, und übertragen auf *VisualWorks*, betrachtet. Weitere Kernpunkte der Arbeit bilden die Möglichkeiten der computergestützten Übersetzung sowie die Beschreibung und Evaluierung des Lokalisierungstools *L10N* im Vergleich zu kommerziell erhältlichen Tools. Das entsprechende Kapitel mit dem Titel "Das Lokalisierungstool *L10N*" soll auch als Informationsquelle für zukünftige Lokalisierungsprojekte mit *L10N* dienen und wurde zu diesem Zweck in die englische Sprache übersetzt.

# 2 Grundlagen

# 2.1 Lokalisierung, Internationalisierung, Globalisierung

Als **Lokalisierung** wird nach Schmitz/Wähle (2000:3) der Prozess der Anpassung eines (Software-) Produktes an regionale Märkte bzw. unterschiedliche Sprach- und Kulturräume bezeichnet, um technisch, sprachlich und kulturell angemessene Programmversionen für die jeweiligen Märkte zu erstellen.

Um ein Softwareprodukt qualitativ hochwertig und effizient lokalisieren zu können, muss das Produkt bereits während seiner Entwicklungsphase internationalisiert werden. Schmitz/Wähle (2000:2) beschreiben **Internationalisierung** als Entwicklung von (Software-) Produkten im Hinblick auf eine leichte Anpassung an andere Märkte, mit dem Ziel, dass Produktversionen für (neue) regionale Märkte schnell und mit möglichst geringem Aufwand erstellt werden können.

Lokalisierung und Internationalisierung werden oft als Bestandteile der **Globalisierung** angesehen.<sup>1</sup> Globalisierung umfasst nach Schmitz/Wähle (2000:2) die Gesamtheit aller Aktivitäten zur internationalen Vermarktung eines Produktes.

Den größten Anteil des Lokalisierungsaufwandes stellt in der Regel die sprachliche Anpassung des Produktes dar. Deshalb ist der folgende Abschnitt übersetzungswissenschaftlichen Grundlagen gewidmet.

Bezugsquelle: Fissgus/Seewald-Heeg 2005:221

# 2.2 Übersetzungswissenschaftliche Grundlagen

In der Übersetzungswissenschaft gibt es unterschiedliche Auffassungen darüber, welche Merkmale eine gute Übersetzung aufweisen sollte. Uneinigkeit herrscht insbesondere darüber, wie stark eine Übersetzung vom Original abweichen darf und welche Freiheiten der Übersetzer hat. Grundsätzlich gilt, dass eine Übersetzung so wörtlich wie möglich und so frei wie nötig sein sollte.

Der Begriff<sup>2</sup> der Äquivalenz spielt in dieser übersetzungswissenschaftlichen Diskussion eine wichtige Rolle. Mit Äquivalenz wird in der Übersetzungswissenschaft allgemein die Gleichwertigkeit von Begriffen bezeichnet<sup>3</sup>. Die Bezeichnung Äquivalenz wird bei Stolze (1999:37f) in Verbindung mit Begriffen unterschiedlicher Sprachen gebraucht, die sich vollständig decken. Decken sich Begriffe unterschiedlicher Sprachen nicht vollständig, werden diese als Entsprechungen bezeichnet.

Für die Bestimmung der Gleichwertigkeit von Begriffen gibt es nach Stolze (1999:38) vier Möglichkeiten. Die Begriffe können vollständig übereinstimmen, dann werden diese auch als kongruent bezeichnet. Ein Begriff kann auch weiter gefasst sein als ein anderer, diesen aber vollständig einschließen. In diesem Fall spricht man von Inklusion. Fehlt ein Begriff in einer anderen Sprache, bezeichnet man dies als Lücke. Die vierte Möglichkeit ist, dass zwei Begriffe unterschiedlicher Sprachen nur teilweise äquivalent sind.

Ein Grund für fehlende oder unvollständige Äquivalenz von Begriffen unterschiedlicher Sprachen liegt darin, dass im fachlichen Diskurs Begriffe häufig konventionell vereinbart werden. Diese sollen das Gelingen der Kommunikation sichern und sie erleichtern. Sie sind aufgrund ihrer Auslegungsbedürftigkeit aber oft strittig und außerdem starken Änderungen unterworfen.

Die fachsprachliche Rede im Deutschen weist generell eine hohe Frequenz von **Fremdwörtern** auf. Häufig sind deutsche Übersetzungen für diese Fremdwörter vorhanden, werden aber von den Fachleuten nicht verwendet. Der Fremdwortgebrauch ist laut Stolze (1999:66) besonders in wissenschaftlichen Darstellungen signifikant, zum einen bei der Verwendung präziser Termini, und zum anderen, weil sich dadurch ein stilistisch höherer

<sup>&</sup>lt;sup>2</sup> "Ein Begriff ist eine Denkeinheit, die diejenigen gemeinsamen Merkmale zusammenfaßt, welche Gegenständen zugeordnet werden." (Arntz/Picht 1995:37)

In engem Zusammenhang dazu steht die Definition von "Benennung":

<sup>&</sup>quot;Eine Benennung ist die aus mindestens einem Wort bestehende Bezeichnung eines Begriffs in der Fachsprache." (Arntz/Picht 1995:37)

<sup>&</sup>lt;sup>3</sup> Bezugsquelle: Gerzymisch-Arbogast 1994:91ff

Anspruch ergibt. Oft sind Anglizismen nach Stolze Fachausdrücke, die in der fachinternen Kommunikation gar nicht anders ausgedrückt werden können.

Ein weiterer wichtiger Aspekt bei der Übersetzung eines Textes ist die **Form der Kommunikation.** Diese kann fachintern, unter Fachleuten eines Faches, interfachlich, unter Fachleuten verschiedener Fächer, oder fachextern, zwischen Fachleuten und Laien, stattfinden. Von der Kommunikationsform hängt es ab, wie viel Vorwissen beim Kommunikationspartner vorausgesetzt wird. Bei der fachinternen Kommunikation soll eine gleiche oder ähnliche kommunikative Funktion in vergleichbarer Situation wie beim Ausgangstext erfüllt werden. Die Übersetzung dient in diesem Fall nur zur Überwindung der Sprachbarriere, ein entsprechendes Fachwissen und die Kenntnis der Fachterminologie werden vorausgesetzt. Bei der fachexternen Kommunikation wird Wissen zwischen der Fachwelt und interessierten Laien ausgetauscht. Solche Texte sollen vor allem verständlich sein.

Vermeer stellte 1978 die **Skopostheorie** auf, nach der jedes translatorische Handeln einem bestimmten Ziel folgt. Demnach werden die Gestaltung des Translationsvorganges und die Form der Translation vom Skopos<sup>4</sup> der Translation bestimmt und die Entscheidung für eine bestimmte Translationsstrategie ist vom Translationsskopos abhängig.

Da das Ziel der Lokalisierung der Benutzeroberfläche von *VisualWorks 7.3* darin besteht, dass Personen mit begrenzten Englischkenntnissen eine deutschsprachige Programmierungebung zur Verfügung gestellt wird, um ihnen das Erlernen des Programmierens zu erleichtern, kann davon ausgegangen werden, dass die Zielgruppe der Übersetzung nur geringe oder keine Fachkenntnisse auf dem Gebiet der Programmierung bzw. der Smalltalk-Programmierung besitzt. Die Kommunikationsform der Übersetzung ist also fachextern und der Hauptanspruch der Übersetzung liegt in der Verständlichkeit des Zieltextes. Diese Forderung nach Verständlichkeit steht im Konflikt mit dem Anspruch, eine Übersetzungslösung zu finden, die den sprachlichen Konventionen der Fachsprache der Programmierer entspricht. Mit Hinblick auf den Translationsskopos des Projektes lag die Priorität bei der Übersetzungsfindung auf der Verständlichkeit der Übersetzung für den deutschsprachigen Benutzer. Deshalb wurde im Allgemeinen den deutschen Entsprechungen der Vorzug vor dem Fremdwortgebrauch gegeben.

<sup>&</sup>lt;sup>4</sup> Griechisch für Ziel, Zweck, Absicht (Metzler Lexikon Sprache 1993:558)

# 3 Internationalisierung

# 3.1 Gründe für die Internationalisierung von Softwareprodukten

Es gibt zwei wesentliche Gründe für die Internationalisierung von Softwareprodukten.

Zum einen wird durch eine gute Internationalisierung sichergestellt, dass das Produkt auf internationalen Märkten **funktional** ist und akzeptiert wird. Um dies zu erreichen, muss das Produkt internationale Zeichensätze, Tastatur-Layouts, fremde Datums- und Zeitformate und fremde Währungen unterstützen. Zur Internationalisierung eines Softwareproduktes gehört auch, dass Online-Hilfe und Dokumentation frei von Jargon, Slang und kulturspezifischen Beispielen sind.

Zum anderen wird durch die Internationalisierung sichergestellt, dass das Softwareprodukt **lokalisierbar** ist. Der Lokalisierungsaufwand für gut internationalisierte Produkte ist in Bezug auf die dafür verwendete Zeit, und somit auch auf die dafür anfallenden Kosten, wesentlich geringer als für schlecht oder gar nicht internationalisierte Produkte.

# 3.2 Folgen unzureichender Internationalisierung

Unterstützt eine Software nur **sprachenabhängige Zeichensätze** wie zum Beispiel ASCII, können in der lokalisierten Version unter Umständen nicht alle Buchstaben, die in der Zielsprache vorkommen, angezeigt werden, oder es werden andere Buchstaben bzw. Zeichen angezeigt als gewünscht. Das Wort  $Gr\u00fc\u00df \u00e9 kann bei Verwendung des ASCII-Codes zum Beispiel nur mit dem$  $erweiterten ASCII-Code f\u00fcr Deutsch auf dem Bildschirm angezeigt werden. Mit der$  $amerikanischen ASCII-Version w\u00fcrden auf dem Bildschirm folgende Zeichen dargestellt$  $werden: <math>Gr_{r}$ -e. Deshalb sollte eine Software, die auf internationalen M\u00e4rkten vertrieben werden soll, den internationalen Standard Unicode unterst\u00fctzen.

Die Übersetzung eines englischen Textes ist meist länger als ihr Original. Wird dies beim Programmieren der Software nicht beachtet, kann dies zur Folge haben, dass die lokalisierte Version **abgeschnittenen Text** enthält, oder dass der Übersetzer Abkürzungen oder Synonyme verwenden muss, die die Qualität des Produktes beeinträchtigen.

Bei der Benennung von Elementen der graphischen Benutzeroberfläche einer Software ist es üblich, kurze Sätze oder einzelne Wörter zu verwenden. Der Programmierer sollte dabei einkalkulieren, dass beim Übersetzen vom Englischen in andere Sprachen ein inverses Verhältnis zwischen der Länge des Ausgangstextes und der Länge der Übersetzung besteht, d. h. je kürzer die englische Zeichenkette ist, umso mehr Platz in Relation zum Ausgangstext wird für die Übersetzung benötigt (vgl. Tabelle 3.1f).

Englisch (Anzahl der Zeichen)	Andere Sprachen (Expansion in Bezug auf
	<b>Original</b> )
1-5	100%
6-20	70%
20-50	30%
>50	15%

Tab. 3.1: Expansion der Textlänge der Übersetzung in Abhängigkeit zur Länge des Originals

(Symmonds 2002:14)

Die folgende Tabelle 3.2 enthält Beispiele aus der Lokalisierung des Programms *VisualWorks 7,3,1*.

Tab. 3.2: Expansion der Textlänge der Übersetzung am Beispiel der Lokalisierung des Programms *VisualWorks* 7.3.1

Englisch		Deutsch	Expansion	
Terminus	Zeichenanzahl	Terminus Zeichenanzah		
Add	3	Hinzufügen	10	233%
Accept	6	Übernehmen	M	67%
Align left	10	Links ausrichten	16	60%
Browse Class/Variable/ Name Space	32	Klasse/Variable/Namens- bereich durchsuchen	41	28%

Der Programmierer sollte den Bildschirm also nicht überladen, damit der Menübaum bei allen Auflösungen auch in den lokalisierten Versionen auf den Bildschirm passt.

Besondere Aufmerksamkeit sollte auch der Programmierung von Datums-, Zeit-, Kalender-, Zahlen- und Währungsformaten geschenkt werden.

Datumsformate können sich bei den Separatoren zwischen ihren Elementen und bei der Reihenfolge ihrer Elemente unterscheiden. So wird in den USA ein Schrägstrich zwischen Tages-, Monats- und Jahresangabe zu deren Abgrenzung verwendet, und in Deutschland werden bei Datumsangaben Punkte als Separatoren eingesetzt. Während in den USA an der ersten Stelle von links meist der Monat angegeben wird, gefolgt von der Angabe des Tages und der Jahresangabe, wird in Deutschland zuerst der Tag angegeben, dann der Monat und das Jahr. Die Angabe 6/10/2005 bedeutet also für einen US-Amerikaner 10. Juni 2005, ein Deutscher würde sie mit 6. Oktober 2005 interpretieren. Es muss also beim Programmieren darauf geachtet werden, dass unterschiedliche Separatoren und Reihenfolgen zugelassen werden.

Auch bei der Angabe von Telefonnummern muss der Programmierer einer Software mehrere Separatoren zulassen, dazu gehören Leerzeichen, Bindestrich, Komma, Punkt und Schrägstrich. In den Vereinigten Staaten von Amerika werden zum Beispiel Bindestriche zur Abgrenzung von Vorwahlen verwendet, in Deutschland werden dagegen Schrägstriche verwendet. Die Dezimaltrennzeichen und die Tausendertrennung von Zahlen können in unterschiedlichen Sprachen ebenfalls voneinander abweichen. Im englischsprachigen Raum wird beispielsweise ein Punkt als Dezimaltrennzeichen verwendet und ein Komma als Tausendertrennung, was im deutschsprachigen Raum umgekehrt ist.

Ebenso muss beachtet werden, dass Währungsformate nicht fest programmiert werden und bei Adressenangaben die Angabe des Landes möglich ist.

In manchen Ländern, wie zum Beispiel in Großbritannien, enthält eine Postleitzahl nicht nur Ziffern, sondern auch Buchstaben. Eine Software, die auf dem britischen Markt vertrieben werden soll, muss an dieser Stelle also auch die Eingabe von Buchstaben zulassen.

Maßangaben müssen auch in anderen Systemen als dem ausgangssprachlichen System angegeben werden können, zum Beispiel muss die Möglichkeit bestehen, Angaben in *inches* auch in *Zentimeter* anzugeben. Das bedeutet, Maßeinheiten dürfen nicht fest kodiert werden.

Einen weiteren wichtigen Aspekt stellt der Umgang mit Symbolen, Grafiken und Farben dar. Diese können in unterschiedlichen Kulturkreisen verschiedene Bedeutungen haben. Die Farbe Rot steht zum Beispiel in den USA und in Westeuropa für Gefahr oder Wut, in China steht sie jedoch für Freude. Das Symbol für den grünen Punkt, der auf dem Papierkorb deutschsprachigen Windows-Version abgebildet ist, stellt einer ein nationalspezifisches Symbol dar, denn er ist nur in Deutschland bekannt. Würde er in anderen Sprachversionen von Windows erscheinen, würde er beim Benutzer eher Verwirrung erzeugen, als die gewünschten Assoziationen hervorzurufen. In Software verwendete Symbole und Grafiken sollten also international bekannt sein, und ihre Wirkung auf das Zielpublikum internationaler Märkte sollte, ebenso wie die Wirkung von Farben, überprüft werden, bevor sie in Software Verwendung finden.

Grafiken sollten außerdem keinen Text enthalten, da dies einen höheren Aufwand bei der Lokalisierung bedeuten würde, denn diese müssen mit speziellen Grafikprogrammen lokalisiert werden.

Zu den regionalen Standards einer Software gehören auch die **Sortierungsregeln**, welche nicht nach einem bestimmten Code, wie zum Beispiel dem ASCII-Code, festgelegt werden sollten. Die Sortierungsregeln sollten vielmehr sprachenspezifisch sein. So sollte zum Beispiel das Zeichen  $\ddot{A}$  im Dänischen am Ende des Alphabetes eingeordnet werden und nicht nach dem Buchstaben *A*. Da in unterschiedlichen Sprachen zum Teil unterschiedliche Buchstaben oder Schriftzeichen verwendet werden, oder die Buchstaben unterschiedlich häufig verwendet werden, gibt es unterschiedliche Tastaturen für unterschiedliche Sprachen. Bei der Entwicklung einer Software muss deshalb sichergestellt werden, dass die Buchstaben für Zugriffstasten auf der für den Zielmarkt spezifischen Tastatur vorhanden sind, alternativ können auch Funktionstasten als Zugriffstasten verwendet werden.

# 3.3 Merkmale eines gut internationalisierten Softwareproduktes

## 3.3.1 Allgemeines

Der Sprachgebrauch in der Originalsoftware hat großen Einfluss auf die Lokalisierbarkeit des Produktes. Generell gilt, es sollten allgemeine Beispiele verwendet werden, die verwendete Terminologie sollte ebenfalls einfach sein und einheitlich verwendet werden, die Sätze sollten kurz sein. Die Zeichensetzung sollte sorgfältig erfolgen, und eine strukturelle Konsistenz sollte gewahrt werden.

#### 3.3.2 Stil

Der sprachliche Stil eines Softwareproduktes sollte dadurch gekennzeichnet sein, dass Mehrdeutigkeiten vermieden werden. Deshalb sollte auf wenig differenzierte englische Verben wie *may, might* oder *can* verzichtet werden. Außerdem sollte der so genannte Telegrammstil vermieden werden, stattdessen sollten die Sätze vollständig ausgeschrieben werden. Auch dies trägt zur Eindeutigkeit von Textteilen bei. Der Satzbau sollte einfach sein, die Sätze sollten kurz und präzise sein und bevorzugt im Aktiv formuliert werden. Trotzdem sollten aber möglichst keine Akronyme<sup>1</sup> und Abkürzungen verwendet werden, da diese in anderen Sprachen oft schwer zu reproduzieren sind. Sollten doch Akronyme verwendet

<sup>,(</sup>auch: Initialwort) Aus den Anfangsbuchstaben oder -silben einer Wortgruppe oder eines Kompositums gebildete Abkürzung, die als Wort verwendet wird [...]" (Metzler Lexikon Sprache 1993:21)

werden, empfiehlt es sich, diese bei ihrer ersten Nennung auszuschreiben.

Der Text sollte ebenfalls wie die verwendeten Symbole und Grafiken nicht kulturspezifisch gefärbt sein, was zum Beispiel durch Jargon, Humor, politische Themen, Verweise auf TV-Sendungen, Umgangssprache, nationale Monumente und ethnische Stereotypen hervorgerufen werden kann. Äquivalente in anderen Sprachen dafür zu finden, kann zwar möglich sein, es ist aber kosten- und zeitaufwändiger, als Äquivalente für neutralen Text zu finden. Außerdem sollten keine Verweise auf Tiere oder Bilder von Tieren verwendet werden, da diese in Asien beleidigend wirken können. Allgemein ist Vorsicht mit Verweisen auf Religionen, heilige Objekte oder heilige Symbole geboten. Ebenso sollte auf Verweise auf Jahreszeiten, Zeitzonen, Wetter und Urlaub, wie zum Beispiel Weihnachtsbäume oder Halloween-Kürbisse, verzichtet werden. Wichtig ist auch, dass bei Produkten für den internationalen Markt keine Vergleiche mit Konkurrenzprodukten angestellt werden, denn dies kann außerhalb der USA unüblich oder sogar illegal sein.

In Deutschland ist der geschäftliche Verkehr beispielsweise durch eine Vielzahl von Gesetzen reglementiert, unter anderem durch das "Gesetz gegen den unlauteren Wettbewerb". Auf einer Internetseite mit dem Titel *Forum Deutsches Recht<sup>2</sup>* kann man dazu Folgendes lesen:

"Das LG Coburg hat entschieden, dass ein Geschäftsmann nachteilige Äußerungen über eine [sie!] Konkurrenten zu unterlassen hat, wenn diese nicht nachweisbar wahr sind. Solche Äußerungen verstossen [sie!] gegen § 14 UWG, da diese Äußerungen geeignet seien, den Betrieb des Konkurrenten zu schädigen. Ein bezifferbarer Schaden sei hierfür nicht erforderlich. Alles, was bei dem Publikum eine nachteilige Wirkung erwecken könne oder den Betrieb erschwere, sei eine Schädigung im Sinne dieser Vorschrift (LG Coburg 23 O 616/99)."

<sup>&</sup>lt;sup>2</sup> http://www.recht.de/index.php3?menue=Rechtsgebiete&linkkat\_id=562&linkkat=152&kname=Wettbewerbsr echt, Zugriff: 06.08.2005

#### 3.3.3 Textexpansion

Um der Textexpansion einer Übersetzung aus dem Englischen in eine andere Sprache Rechnung zu tragen, sollte die Größe der Elemente einer grafischen Benutzeroberfläche einer Software so gewählt werden, dass genügend Platz für diese längeren Übersetzungen vorhanden ist. Dies gilt insbesondere für Tabellen, Diagramme und Illustrationen. Auch die vertikale Ausdehnung bei Doppelbyte-Sprachen wie Japanisch oder Chinesisch darf nicht vergessen werden. Zwischen Sätzen und nach allen Satzzeichen sollte nur ein Leerzeichen gewählt werden, wie es in den meisten Sprachen üblich ist, und nicht zwei Leerzeichen, wie es im englischsprachigen Raum teilweise verbreitet ist<sup>3</sup>.

# 3.4 Kontrollierte Sprache

Kontrollierte Sprache ist eine Teilmenge einer natürlichen Sprache. Sie wird unter anderem verwendet, um technische Dokumentation zu schreiben, üblicherweise für eine spezielle Firma oder einen speziellen Industriezweig. Ihre Grundkomponenten können Termini für bestimmte Konzepte und Grammatikregeln sein, ebenso wie Wortlisten und Redewendungen, die vermieden werden sollten, zusammen mit entsprechenden Synonymen. Ihre Vorteile bestehen darin, dass der Text weniger Mehrdeutigkeiten enthält und besser lesbar ist. Außerdem ist es mit kontrollierter Sprache einfacher, Konsistenz im Stil und bei der Verwendung der Terminologie zu wahren. Eine bessere Übersetzbarkeit kann mit kontrollierter Sprache ebenfalls erreicht werden, besonders bei Verwendung von maschinellen Übersetzungssystemen und bei computergestützter Übersetzung. Durch die genannten Vorteile kann der Zeitaufwand für die Übersetzung vermindert werden, und das Produkt kann demzufolge schneller auf den Markt gebracht werden.

Der Einsatz kontrollierter Sprache kann durch spezielle Softwareanwendungen unterstützt werden. Diese unterbreiten meist Vorschläge für Grammatik und Vokabular und/oder beschränken das Vokabular und die Satzstrukturen. Dies kann zum Beispiel eine beschränkte Wortanzahl pro Satz sein, Einschränkungen bei den Zeitformen, oder die Einschränkung, dass kein Passiv verwendet werden darf.

<sup>&</sup>lt;sup>3</sup> Bezugsquelle: http://en.wikipedia.org/wiki/Full\_stop#Spacing\_after\_full\_stop, Zugriff: 06.08.2005

# 3.5 Externalisierung der übersetzbaren Daten

Um ein qualitativ hochwertiges Lokalisierungsergebnis zu erzielen und die Kosten dafür zu minimieren, sollte der Programmtext in einen Code-Block und in einen Daten-Block unterteilt werden, wobei der Code-Block den Programmcode enthält und der Daten-Block alle übersetzbaren Daten. Dies dient zum einen der Effizienz, denn für eine neue Sprachversion des Programms braucht in diesem Fall lediglich der Daten-Block übersetzt werden, und zum anderen bietet diese Vorgehensweise ein hohes Maß an Sicherheit, da der Programmcode unberührt bleibt und nicht während des Übersetzungsprozesses beschädigt werden kann. Zusätzlich wird dadurch die Wahrscheinlichkeit verringert, dass übersetzbarer Text bei der Übersetzung übersehen wird, und der Aufwand für den Funktionstest in der Qualitätssicherungsphase kann deutlich verringert werden.

## 3.6 Internationalisierung von VisualWorks

# 3.6.1 Allgemeines

Die Externalisierung der übersetzbaren Daten wird bei *VisualWorks* mithilfe von so genannten "Locales" realisiert. In der Online-Enzyklopädie *Wikipedia* wird *Locale* folgendermaßen definiert:

"Eine **locale** [sie!] ist ein Einstellungssatz, der die Gebietsschemaparameter für Computerprogramme enthält." (http://de.wikipedia.org/wiki/Locale, Zugriff: 02.08.05)

Eine Locale enthält Informationen, die für die Anpassung einer Anwendung an geographische oder kulturelle Normen notwendig sind, wie zum Beispiel die Zeichencodierung, den Schriftgrad und die Schriftart, die Sortierungsregeln für die Zeichenketten sowie Formate für Datum, Zeit oder Währung.

In *VisualWorks* werden Instanzen der Klasse Locale verwendet um unterschiedliche Locales wiederzugeben. Der Name der Locale setzt sich aus jeweils einem Kürzel für die verwendete Sprache, dem Land und der verwendeten Codierung zusammen. Dabei wird die Sprachenangabe von einem Unterstrich gefolgt und das Land in Versalien angegeben, gefolgt von einem Punkt.

Die Locale für amerikanisches Englisch könnte auf einer HP-Maschine beispielsweise wie folgt aussehen: #'en\_US.roman8'. Die Bezeichnung der Sprache ist dabei nach der ISO-Norm 639 normiert und die Landesbezeichnung nach ISO 3166 alpha-2.

Beim Starten von *VisualWorks* wird das Betriebssystem abgefragt, welche Locale genommen werden soll, dann wird eine passende Locale-Instanz von den im Image definierten Locale-Instanzen ausgewählt. Ist keine oder keine passende Instanz vorhanden, wird standardmäßig die so genannte C-Locale gewählt, welche benannt wurde nach dem dem Anspruch nach kulturneutralen Englisch der Programmiersprache C.

#### 3.6.2 Funktionsweise von Nachrichtenkatalogen

Eine Softwareanwendung hat meist viele unterschiedliche Textelemente auf ihrer Oberfläche, wie zum Beispiel Labels<sup>4</sup>, Eingabefelder, Menü-Elemente und Dialogfelder. Text, der vom Benutzer der Anwendung gesehen wird, wird als Benutzernachricht bezeichnet.

Eine Anwendung mit mehreren Locales erfordert für jede Benutzernachricht pro Locale eine eigene Übersetzung. In *VisualWorks* wird dies mittels Nachrichtenkatalogen realisiert.

Die Funktionsweise von Nachrichtenkatalogen soll am Beispiel eines Labels beschrieben werden, welches in den Sprachen Englisch, Französisch und Spanisch verfügbar gemacht werden soll. Dazu wird dem Label ein Suchschlüssel zugewiesen, der das passende Label in Abhängigkeit von der aktuellen Locale sucht. Wird die Anwendung mit einer englischen Locale ausgeführt, wird in einem englischen Katalog gesucht, bei einer französischen Locale in einem französischen Katalog und so weiter. In Abbildung 3.1 ist die Funktionsweise von Nachrichtenkatalogen schematisch dargestellt.

<sup>&</sup>quot;Ein Bezeichner. [...] Es kann sich auch um ein Wort, ein Symbol oder eine andere Zeichenfolge handeln, die man zur Identifizierung einer Datei, eines Speichermediums, eines in einem Computerprogramm definierten Elements oder eines spezifischen Elements in einem Dokument (beispielsweise einem Tabellenblatt oder Diagramm) verwendet." (MS Press Title 2000)



Abb. 3.1: Funktionsweise von Nachrichtenkatalogen<sup>5</sup>

Ein Nachrichtenkatalog ist also eine Liste, die aus Suchschlüsseln mit einer Buchstabenzeichenkette für jeden einzelnen Schlüssel besteht.

### 3.6.3 Mehrere Kataloge pro Locale verwenden

Es können entweder alle Suchschlüssel und Werte in einer einzelnen Datei definiert werden, oder auch bausteinartige Dateien erstellt werden. Kleinere Dateien haben dabei den Vorteil, dass sie einfacher zu pflegen sind und die Suche schneller ausgeführt werden kann.

Ein Beispiel dafür sind Kernanwendungen mit mehreren optionalen Modulen. Durch einen separaten Katalog für jedes Modul und jede Locale ist es möglich, jedem Anwendungsbenutzer nur die Kataloge anzubieten, die für diese Konfiguration von Anwendungsmodulen erforderlich sind.

<sup>&</sup>lt;sup>5</sup> Bezugsquelle: Cincom VisualWorks Internationalization Guide, http://smalltalk.cincom.com/prodinformation/index.ssp?content=vwdocumentation, Zugriff: 04.09.2005

### 3.6.4 Die Suche mit mehreren Katalogen optimieren

Bei großen Anwendungspaketen kann es vorkommen, dass für jede Locale sehr viele Nachrichtenkataloge existieren. Für jede einzelne Benutzernachricht wird bei Bedarf jeder einzelne Katalog durchsucht, um die entsprechende Zeichenkette zu finden. Um die Suche einzugrenzen, kann man für jede Benutzernachricht nur in bestimmten Katalogen suchen lassen. Dazu muss zu der Benutzernachricht der Name des ihr zugeordneten Kataloges (Katalog-Kennzeichnung) zugewiesen werden.

Nachrichtenkatalog-Dateien können beliebig organisiert werden, zum Beispiel nach Art der Nachricht (Dialogfelder, Label, usw.). Es ist dabei jedoch zu berücksichtigen, dass sich die Zahl der Dubletten umso mehr erhöht, je mehr Unterteilungen vorgenommen werden. Die Vorteile kleinerer Dateien müssen gegen die Kosten, die durch Dubletten entstehen, abgewogen werden, und es sollte ein Mittelmaß gefunden werden.

### 3.6.5 Fehlgeschlagene Suche

Wird ein Suchschlüssel nicht in den Katalog-Indizes gefunden, wird dieser Suchschlüssel in eine Zeichenkette konvertiert. Diese Zeichenkette wird dann anstelle des Suchergebnisses zurückgegeben. Es kann aber auch eine Standardzeichenkette erstellt werden, die bei fehlgeschlagener Suche ausgegeben wird.

# 4 Computergestützte Übersetzung

# 4.1 Allgemeines

Obwohl eine qualitativ hochwertige vollautomatische Übersetzung zum heutigen Zeitpunkt nicht möglich ist, kann computergestützte Übersetzung für bestimmte Textsorten und Verwendungszwecke als sinnvolles Hilfsmittel bei der Übersetzung eingesetzt werden. Zum Beispiel wenn es sich um Texte mit häufigen Wiederholungen von Termini und/oder syntaktischen Strukturen handelt, wie es zum Beispiel bei Bedienungsanleitungen bzw. Gebrauchsanweisungen der Fall ist.

Bei computergestützter Übersetzung wird - wie in Abbildung 4.1 dargestellt - zwischen maschinengestützter Übersetzung (Machine-Aided Human Translation - MÄHT) und maschineller Übersetzung (Human-Aided Machine Translation - HAMT bzw. Fully Automatic Machine Translation - FAMT) unterschieden.



Abb. 4.1: Kategorien der computergestützten Übersetzung

Bei der maschinengestützten Übersetzung erfolgt die eigentliche Übersetzungsarbeit durch einen menschlichen Übersetzer, der durch Softwaresysteme unterstützt wird. Das kann zum Beispiel durch automatisches Nachschlagen von Termini geschehen, oder durch das Zerlegen von Texten und Sätzen in so genannte Übersetzungseinheiten, welche auch als Segmente bezeichnet werden können. Außerdem werden von einem solchen System Übersetzungsvorschläge angeboten, die aus früheren Übersetzungen stammen und in den Übersetzungsspeicher, das so genannte Translation Memory, eingespeist wurden.

Beispiele für kommerziell erhältliche Translation-Memory-Systeme sind zum Beispiel die Translator's Workbench von Trados, Transit von Star, Dejä Vu von Atril, SDLXvon SDL International oder Wordfast von Champollion.

Bei der **maschinellen Übersetzung** wird die Übersetzung von der Software ausgeführt. Es wird unterschieden zwischen vollautomatischer maschineller Übersetzung (FAMT) und maschineller Übersetzung in Verbindung mit menschlicher Interaktion (HAMT).

Im zweiten Fall obliegt dem menschlichen Übersetzer die Vorbereitung der zu übersetzenden Texte, das Pre-Editing, und die Nachbereitung des von der Software erstellten Übersetzungsergebnisses, das Post-Editing.

Ein wichtiger Aspekt beim Pre-Editing ist der Einsatz von so genannter kontrollierter Sprache. Unter kontrollierter Sprache versteht man eine natürliche Sprache, welche explizit definierten Beschränkungen in Lexik, Grammatik und Stil unterliegt. Sie dient der Reduzierung von Ambiguitäten<sup>1</sup>. Regeln für ein kontrolliertes Englisch könnten zum Beispiel darin bestehen, die maximale Wortanzahl eines Satzes zu beschränken, kein Passiv zu verwenden oder bestimmte Lexik zu verwenden bzw. nicht zu verwenden.

Ein maschinelles Übersetzungssystem führt als ersten Schritt eine syntaktische Analyse durch, auf deren Grundlage anschließend die ausgangssprachlichen Lexeme<sup>2</sup> durch zielsprachliche Entsprechungen ersetzt werden. Darauf basierend, und unter Anwendung syntaktischer und morphologischer Regeln der Zielsprache, wird dann der Zieltext generiert.

Beispiele für kommerziell erhältliche maschinelle Übersetzungssysteme sind Systran, LOGOS oder der Personal Translator von Linguatec.

<sup>&</sup>quot;(lat. ambigutias >Doppelsinn<. Auch: Ambivalenz, Amphibolie, Mehrdeutigkeit, Vieldeutigkeit) [...] Die Mehrdeutigkeit eines Einzelzeichens, d. h. eines Lexems, einer Ableitung, eines Grammems oder Kompositums [...]." (Metzler Lexikon Sprache 1993:35)

<sup>&</sup>lt;sup>2</sup> "Einheit des Wortschatzes, die die begriffliche Bedeutung trägt." (Duden 2001)

Viele Translation-Memory-Systeme besitzen eine Funktion, bei der der zu übersetzende Text mit dem Inhalt eines Übersetzungsspeichers verglichen wird, und übereinstimmende Teiltexte anschließend durch die im Übersetzungsspeicher vorhandenen Übersetzungen ersetzt werden können. Zum Teil werden solche Funktionen als "automatische Übersetzung" oder "Vorübersetzung" bezeichnet. Eine solche Ersetzungsfunktion kann nicht mit maschineller Übersetzung gleichgesetzt werden.

# 4.2 Translation-Memory-Systeme

#### 4.2.1 Allgemeines

Die Idee, die dem Einsatz von Translation-Memory-Systemen zugrunde liegt, besteht darin, vorhandene Übersetzungen zusammen mit ihrem Ausgangstext für Übersetzungen ähnlicher Texte verfügbar zu machen. Ein Translation-Memory-System ist eine integrierte Übersetzungsumgebung mit automatischem Zugriff auf vorhandene Übersetzungseinheiten und deren Übersetzungen, welche in einem so genannten Translation Memory, auch Übersetzungsspeicher genannt, abgelegt sind, und auf terminologische Daten, die in einem Terminologieverwaltungssystem abgelegt sind. Die vom System automatisch vorgenommene Untergliederung in Übersetzungseinheiten, auch Segmentierung genannt, erfolgt meist satzweise.

In den folgenden Abschnitten werden die grundlegenden Funktionen von Translation-Memory-Systemen am Beispiel von *Trados* und *SDLX* erläutert.<sup>3</sup>

<sup>&</sup>lt;sup>3</sup> Für die Diplomarbeit verwendete Versionen: *Trados 6.5 Freelance, SDLX 2004 Standard* 

#### 4.2.2 Der Übersetzungseditor

Es wird zwischen zwei Arten von Übersetzungseditoren unterschieden. Eine Möglichkeit besteht in der Verwendung von systemeigenen Editoren, in denen die Ausgangstexte, die in unterschiedlichen Formaten vorliegen können, geöffnet werden, nachdem sie in ein vom Editor lesbares Format überführt wurden. Beispiele hierfür sind *Transit* und *SDLX* (vgl. Abbildung 4.2).



Abb. 4.2: Der Übersetzungseditor von SDLX

Für diese Konvertierung werden die Layout-Informationen des Ausgangstextes in Steuerzeichen (Tags) umgewandelt. Diese sind während des Übersetzungsvorganges schreibgeschützt. Der Schreibschutz kann bei Bedarf jedoch vom Übersetzer aufgehoben werden, und dann können diese Tags auch vom Übersetzer bearbeitet werden. Nachdem der Übersetzungsvorgang abgeschlossen wurde, wird das ursprüngliche Format des Ausgangstextes wieder hergestellt.

Als zweite Möglichkeit kann auch ein gängiges Textverarbeitungssystem als Editor genutzt werden. Ein Beispiel hierfür ist die *Translator's Workbench*, welche mit *Microsoft Word* verwendet werden kann (vgl. Abbildung 4.3). Dabei wird direkt im Originaldokument übersetzt. Bei der *Workbench* wird die aktuelle Übersetzungseinheit blau hinterlegt, und das Feld für die Übersetzung grün (100%-Match) bzw. gelb (Fuzzy-Match<sup>4</sup> oder kein Match).



Abb. 4.3: Die Übersetzungsumgebung der Translator's Workbench mit Word

### 4.2.3 Die Terminologieverwaltung

Die Terminologieverwaltungskomponente enthält eine Schnittstelle zum Übersetzungseditor bzw. zum Textverarbeitungssystem. Bei einigen Translation-Memory-Systemen wird die Terminologiedatenbank automatisch nach Termini durchsucht, die in der aktuellen Übersetzungseinheit vorkommen. Das ist zum Beispiel bei der *Translator's Workbench* und bei *SDLX* der Fall.

Eine andere Möglichkeit besteht darin, dass die Terminologiesuche manuell über eine Schaltfläche, eine Tastenkombination oder über die Menüleiste gestartet wird. In Abbildung 4.4 ist beispielsweise das Ergebnis einer manuellen Terminologiesuche mit *Dejä Vu* dargestellt.

IAlle Zeilen		- De	ubseta		Calephy di	to:	1
nzösisch	Deutsch "				A 01 2		_
Le Monde.fr :	Le Monde.f	r:				wtwj.lemande.fr.htm	
A la Une	Auf der Tit	ElseitE					
INDEX, FOLLOW, NOARCHIVE							
LE MONDE. Journal Le Monde, quotidien d'information	LE MONDE.	🖳 Terminolog	jiesuche			×	1
francophone / Le Monde, the french quality newspaper of	franzÖSisd	Überein5timmur	ngen:				
record	quality nev	Französisch		Deutsch		້ ເງິນ ເມື່ອງ ເມື່ອງ ເມື່ອງ ເພື່ອງ ເພື່	
0		Netscape		Netscape		Schließen	
NEWS, PRESSE, PRESS, NEWS, RRANCE, FRENCH, DOSSIERS, ECONOMIE, ECONOMY, CULTURE, INTERNATIONAL, BOURSE, CINEMA, MOVIES, LIVRES, BOOKS, MULTIMEDIA, EDUCATION, FORUMS, FORUM, SERVICES, ABONNEMENTS, BOUTIQUE, EMPLOI, EXPOSITIONS, FESTIVALS, SPORT, MAGAZINE, EUROPEEN, DIPLOMATIQUE, PARTENAIRES, PUBLICITE, LETTRES D'INFORMATIONS, NEWSLETTERS, JOURNAL EN LIGNE, LE MONDE ON LINE, VERSION PALM, VERSION MOBILES, MOBILE SERVICES, METEO, ARCHIVES, DOCUMENTATION, NOUVELLES TECHNOLOGIES, HIGH TECH, TRADUCTEUR, TRANSLATOR	NEWS, PRI DOSSIERS INTERNAT: BOOKS, MI SERVICES, AUSTELLUI EUROPÄIS INFORMAT TAGESZEF VERSION & ARCHIVE, I TECH, ÜBE	ftusgangstexta Context	ttribute: Netscape	Zieltexteigens Context	schaften: Netscape	Einfügen ^< Einfügen ^> Kopieren	
Netscape							
expires-							
path=		Dotum: 11.0	2 2005 22:16:28	Poputzor: Ar	dministrator		
path=		Projekt:	2	Datei: w			
domain=		Kunde:	Conoralition				
secure		Datenbank:	D:\Daten\HA\see	wald\7_sem\sdv	Belegarbeit/lemonde	e\tbdejavu\tbdejavu.c <sup>1</sup>	
- 1-					1	. 175	

Abb. 4.4: Ergebnis einer manuellen Terminologiesuche mit Dejä Vu

Die gefundenen Termini und deren Übersetzung werden in einem eigenen Fenster angezeigt, und die Übersetzung kann dann direkt, zum Beispiel über eine Tastenkombination, in die Übersetzung eingefügt werden.

#### 4.2.4 Das Translation Memory

Das Translation Memory ist der Speicherort für die ausgangssprachlichen Übersetzungseinheiten und deren Übersetzungen.

Während des Übersetzungsvorganges wird die aktuelle Übersetzungseinheit automatisch im Übersetzungsspeicher gesucht und gegebenenfalls zusammen mit der dazugehörigen Übersetzung, ebenso wie die gefundenen Termini, in einem gesonderten Fenster angezeigt. Die Übersetzung kann ebenfalls direkt in den aktuell zu übersetzenden Text eingefügt werden.

Der Abgleich der aktuellen Übersetzungseinheit mit dem Translation Memory ist nicht nur auf Segmente beschränkt, die mit dem Suchtext zu einhundert Prozent übereinstimmen, sondern es können auch ähnliche Segmente gesucht werden, die nur zu Teilen mit dem aktuellen Segment übereinstimmen. Eine solche Suche wird als Fuzzy-Match-Suche bezeichnet. Der so genannte Matchwert wird basierend auf der Ähnlichkeit der Zeichenketten ermittelt. Der Benutzer hat die Möglichkeit, festzulegen, wie hoch der Matchwert eines Suchergebnisses sein muss, damit es als Übersetzungsvorschlag (Match) angezeigt wird (vgl. Abbildung 4.5).

Undo ]	Spell Checker	] Auto	-propagate Editor		Automation
Format Paint	Translation Mem		toMatch ]	lviachi	ne Translation
Minimum rr	natch value (Z)			m	źĴ
Maximum r	umber of hits			4	-
Content Mi	smatch Penalty			1	H
Filter Misma	atch Penalty			1	÷
F~ Open t	he last TMs at sta	rtup			
r" Hide Sy	stem fields in toolt	ips			
F~ Enable	multiple TM acce	SS			

Abb. 4.5: Einstellung des minimalen Matchwertes, der Anzahl der angezeigten Treffer und der Matchwert-Abzüge bei *SDLX* 

Bei vielen Systemen kann der Benutzer auch selbst Faktoren festlegen, die zu Abzügen beim Matchwert führen. Zum Beispiel wenn die Quelle eine maschinelle Übersetzung ist, oder wenn es Unterschiede in der Formatierung gibt (vgl. Abbildung 4.6).

		provide a	1	
llgennein At	ozüge   Einstellungen für Ersetzungen   Konko	rdanz j E	xtras	
	Abzug für unterschiedliche Formatierung X:	JŪ	·	
Abzug	für unterschiedliche Attribute und Textfelder X:	12		
	Abzug für unterschiedliche Placeables X:	12	-]	
	Abzug für Alignrnent X:	3		
	Abzug für maschinelle Übersetzung X:	15		
	Abzug für mehrere 1 Ü0%-Matches Ä:	fÖ		
A	bzug für Placeables auch bei anderen Tags im usgangssegment	F		
		OK	Abbrechen	Hilfo

Abb. 4.6: Einstellung der Matchwert-Abzüge bei der Translator's Workbench

Die Unterschiede zum aktuellen Segment werden vom System zum Beispiel - wie die Abbildungen 4.7 und 4.8 zeigen - durch farbliche Hervorhebung kenntlich gemacht.



Abb. 4.7: Anzeige der Abweichungen bei der Fuzzy-Match-Suche von SDLX



Abb. 4.8: Anzeige der Abweichungen bei der Fuzzy-Match-Suche der Translator's Workbench

Bei der so genannten Konkordanzsuche, auch Kontextsuche genannt, werden alle Kontexte angezeigt, in denen das gesuchte Wort bzw. die gesuchte Zeichenkette vorkommt. Die Konkordanzsuche findet unabhängig vom Abgleich des aktuell zu übersetzenden Textes mit dem Übersetzungsspeicher statt. Sie kann zum Beispiel verwendet werden, wenn kein Match oder kein passendes Match im Translation Memory gefunden wurde. Dadurch kann der Übersetzer sehen, in welchen Kontexten ein Wort bzw. eine Wortgruppe im Ausgangstext oder im Zieltext verwendet wurde (vgl. Abbildung 4.9f).

bultering.itd - SDL Edit	<u>_181</u>
a Edit Format Format Paint View Tools Translation Memory TermBase Help	o
L M C M M M M M M M M M M M M M M M M M	
2:\dipliomarbeitAsdliAbuffer.mdb	
DAdiplomaibeiftsdiiAbLfcing.tdb	
Avoiding Buffer Underruns Vermei	den von Pufferunterschreitung
<sup>2</sup> Data buffering is the process in which the clata tobewrittentothe CD is storedin atemporary location betöre being sentto the CD-R/RW drive. <sup>2</sup> Das S Puffers Daten soll, in «erden gesen	peichern von Daten in den speichern von Daten in den speicher ist der Prozess, bei dem die , mit denen die CD beschrieben werden einer temporären Datei gespeichert n, bevor sie an das CD-R/RW-Laufwerk det werden.
<sup>3</sup> All CD-R/RWdnves absolutely require that data written to the CD be streamed to the drive in one continuous flow. Cü-R/F	aten, die auf die CD geschrieben n, müssen in jedem Fall in einem uierlichen Datenfluss zu dem RW-Laufwerk geleitet werden.
<sup>4</sup> If this data stream is interrupted, for any	lieser Datenstrom aus irgendeinem
<sup>4</sup> If this data stream is interrupted, for any Ion* I vermeiden von Pufferunterschreitung	ieser Datenstrom aus irgendeinem
<sup>4</sup> If this data stream is interrupted, for any <sup>4</sup> Wird d Ion* I vermeiden von Pufferunterschreitung	ieser Datenstrom aus irgendeinem
If this data stream is interrupted, for any     Ivermeiden von Pufferunterschreitung	<sup>1</sup> Vermeiden von Pufferunterschreraung <sup>2</sup> Es gibt viele Möglichkeiten fUr den Benutzer classibeizutragen, dass die Pufferunterschreitung vermieden wird.
<ul> <li><sup>4</sup> If this data stream is interrupted, for any <sup>4</sup> Wird d</li> <li>Ion<sup>*</sup> I vermeiden von Pufferunterschreitung</li> <li><sup>1000</sup> <sup>1</sup> Avoidmg Buffer underruns</li> <li><sup>1000</sup> <sup>1</sup> Theorem are interrupted in the stream and the drive runs out of data, a buffer underrun occurs and the disc is usually rendered unusable</li> </ul>	<ul> <li><sup>1</sup> Vermeiden von Pufferunterschreraung</li> <li><sup>2</sup> Es gibt viele Möglichkeiten fUr den Benutzer classibeizutragen, dass die Pufferunterschreitung vermieden wird.</li> <li><sup>3</sup> In diesem Fall tritt eine Pufferunterschreitung auf und die C0 kann dann für gewöhnlich nicht mehr verwendet werden</li> </ul>
<ul> <li><sup>4</sup> If this data stream is interrupted, for any <sup>4</sup> Wird d</li> <li>Interrupted for any <sup>4</sup> Wird d</li> <li>Interrupted for any <sup>4</sup> Wird d</li> <li>Interrupted for an experimental stream of the stream</li></ul>	<ul> <li><sup>1</sup> Vermeiden von Pufferunterschreraling</li> <li><sup>2</sup> Es gibt viele Möglichkeiten fUr den Benutzer classibeizutragen, dass die Pufferunterschreitung vermieden wird.</li> <li><sup>3</sup> In diesem Fall tritt eine Pufferunterschreitung auf und die C0 kann dann für gewöhnlich nicht mehr verwendet werden</li> <li><sup>4</sup> Deshalb werden die Daten üblicherweise in einem Pufferspeicher abgelegt, bevor sie zu dem Lauftvek gesendet werden.</li> </ul>
<ul> <li><sup>4</sup> If this data stream is interrupted, for any <sup>4</sup> Wird d</li> <li>Interrupted on the stream is interrupted, for any <sup>4</sup> Wird d</li> <li>I vermeiden von Pufferunterschreitung</li> <li><sup>1</sup>Avoidmg Buffer onderruns</li> <li><sup>2</sup> The stream is interrupted unusable</li> <li><sup>2</sup> The stream is usually rendered unusable</li> <li><sup>3</sup> When the drive runs out of data, a buffer underrun occurs and the disc is usually rendered unusable</li> <li><sup>4</sup> Wird d</li> <li><sup>5</sup> This buffering allows the Software to account for small changes in the data stream, which may be caused by any InurTHÜfcofexternal factors.</li> </ul>	<ul> <li><sup>1</sup> Vermeiden von Pufferunterschreräung</li> <li><sup>2</sup> Es gibt viele Möglichkeiten fUr den Benutzer classibeizutragen, dass die Pufferunterschreitung vermieden vird.</li> <li><sup>3</sup> In diesem Fall tritt eine Pufferunterschreitung auf und die Co kann dann für gewöhnlich nicht mehr verwendet werden</li> <li><sup>4</sup> Deshalb werden die Daten üblicherweise in einem Pufferspeicher abgelegt, bevor sie zu dem Lauftvek gesendet werden.</li> <li><sup>5</sup> Durch diesen Speichervorgang kann die Software klein AÄnderungen im Datenstrom eliminieren, die durch mehrere eixterne Faktoren verursacht werden können</li> </ul>

Abb. 4.9: Konkordanzsuche bei SDLX

Suchen CCZurück Weiteroo	Löschen Dptionen.	Schließe
ata	2	Hife
All CD-R/RW drives absolutely reguire the	it data written to the CD be streamed to the drive in one continuous flow.	
1 Es ist bei allen CD-R/RW-Laufwerken ur	bedingt erforderlich, dass die Daten, die auf die CD geschriebenen werder	n, in einem kontinuierlichen Datenfluss zu dem
Laufwerk geleitet werden.		
Angelegt am: 18.07.2005,11:18 Angelegt von:	JSER Geändeitam: Geändeitvon:	
If this data stream is interrupted.for any	reason.the potentialforthe drive to essentially 'run out'of data is very hig	h.
1 Wird dieser Datenstrom aus irgendeinem	Grund unterbrochen, kann es sehr schnell passieren, dass das Laufwerk	k keine Daten mehr zur Verfügung hat.
Angelegt am: 1S. 07.2005,11:18 Angelegl von	JSER Geändeitam: Geändertvon:	
When the drive runs out of data, a buffer	underrun occurs and the disc is usually rendered unusable.	
5 Wenn das Laufwerk keine Daten mehr z	ur Verfügung hat, tritt eine Unterschreitung des Pufferspeichers auf und die	e CD wird für gewöhnlich unbrauchbar gemacht.
Angelegt am: 1S. 07.2005,11:18 Angelegt von	JSER Geändertam: Geändertvon:	
1 This is why data is usually buffered befor	e it is sent to the drive.	
1 Deshalb werden die Daten üblicherweise	in einem Pufferspeicher gespeichert, bevor sie zu dem Laufwek gesendet	t werden.
Angelegt am: 18.07.2005,11:18 Angelegt von	JSER Geändertam: Geändertvon:	
S This buffering allows the Software to acco	ount for small changes in the data stream, which may be caused by any r	number of external factors.
1 Durch diesen Speichervorgang kann die	Software kleine Änderungen im Datenstrom eliminieren, die durch mehrere	e externe Faktoren verursacht werden können.
Angelegtam. 18.07.2005,11:18 Angelegt von I	JSER Geändertam: Geändert von-	
As a result, the flow of data to the drive i	nay be interrupted.	
1 Aufgrund dessen kann der Datenfluss zu	m Laufwerk unterbrochen werden.	
Angelegtam 18.07.2005,11:18 Angelegt von	JSER Geändeitam- Geändeitvon-	
1 Your data access is no longer held host	age by machine dependence or location.	
Ihr Datenzugriff keine Geisel mehr von N	aschinenabhängigkeit oder Ortsgebundenheit.	
Angelegtami: 18.07.2005,11:18 .Angelegtvon.	JSER Geändeitam: Geändeitvon:	
1 Yourwork environment now revolves arou	nd you, and your data is with you wherever you are.	
Ihre Arbeitsumgebung dreht sich von nun	an um Sie und Ihre Daten sind bei Ihnen, egal wo Sie sind.	
Angelegtami: 18.07.2005,11:18 Angelegt von:	JSER Geändeitam: Geändert™-	
R		

Abb. 4.10: Konkordanzsuche bei der Translator's Workbench

Die meisten Systeme sind auch in der Lage, zusätzliche Informationen, wie den Namen des Übersetzers, das Anlegedatum, das (letzte) Änderungsdatum oder die verwendete Quelle, im Übersetzungsspeicher abzulegen.

Ein Übersetzungsspeicher ist zu Beginn einer Übersetzung leer. Er kann gefüllt werden, während eine Übersetzung erstellt wird.

Bei der *Translator's Workbench* geschieht dies durch den Befehl **Speichern und** nächsten Satz öffnen, welcher über eine Schaltfläche, eine Tastenkombination oder über die Menüleiste ausgewählt werden kann.

Bei *SDLX* können über den Befehl **Update**, welcher ebenfalls über eine Schaltfläche, eine Tastenkombination oder die Menüleiste ausgewählt werden kann, alle Übersetzungseinheiten, die den Status **confirmed** (geprüft) besitzen, in das Translation Memory gespeichert werden.

Eine weitere Möglichkeit, den Übersetzungsspeicher zu füllen, stellt der Import eines bereits vorhandenen Translation Memory dar. Damit dies mit möglichst wenig Aufwand und Datenverlusten erzielt werden kann, wurde das Austauschformat TMX (Translation Memory Exchange) entwickelt. Hierbei handelt es sich um einen herstellerunabhängigen XML-Standard, welcher durch die Arbeitsgruppe *OSCAR* (Open Standards for Container/Content Allowing Re-use), ein internationales Konsortium von Herstellern und Großanwendern von Übersetzungstools innerhalb der *LISA* (Localization Industry Standards Association - ein Interessenverband von Unternehmen mit Tätigkeit in den Bereichen Globalisierung, Internationalisierung, Lokalisierung und Übersetzung), entwickelt wurde. Für den Austausch von Terminologie-Datenbanken wurde von derselben Organisation das Austauschformat TBX (Termbase Exchange) entwickelt.

Eine dritte Möglichkeit, einen Übersetzungsspeicher zu füllen, bietet die Alignment-Funktion vieler Translation-Memory-Systeme. Mit Alignment bezeichnet man den Vorgang der Paarbildung von äquivalenten Segmenten eines Ausgangstextes und eines Zieltextes, die in maschinenlesbarer Form vorliegen.

Bei einem Alignment werden die Texte als erstes vom System in Segmente unterteilt. Anschließend nimmt das System, hauptsächlich aufgrund der Berechnung der Länge der Zeichenketten, eine paarweise Zuordnung der ausgangssprachlichen und zielsprachlichen Segmente vor (vgl. Abbildung 4.11f).

🔡 alignment sicherheit.iad – SDL Align	_isi×
File Edit View Alignment Tools Translation Memory Help	
st 4iela= >> 1K> >> >> >> >> >> >> >> >> >> >> >> >> >	GUO 3 INI
AMögliche Gefahren (durch das Lösungsmittel DK)	13 ADaprers qui pouvent apparaitre (par le disselvant DK)
<sup>2</sup> Prophere Elügeigkeit	<sup>2</sup> il iguide inflommable
<sup>3</sup> Ilroversibler Sebadan mäglich	<sup>3</sup> Dommoro irrovorsible possible
<sup>1</sup> Cooundhaiteachädlich: kapp haim Verschlucken und enschlie	<sup>4</sup> Insolubre : pout course des offections du pourses si en l'ovale
<sup>5</sup> Cosundhoiteschödliche Zubereitung im Sinne PL 1000/45/EC	<sup>5</sup> Reparation incolubro au conc. PL 1000//5/EC
<sup>6</sup> :4 AEroto Hilfo Mo(nohmon	<sup>6</sup> 4 Moguros do promior accours
Allogmaina Hinwaisa: Friesbluft kontaminiata Klaidung antfar	<sup>1</sup> IPonocianomento générouve or frois, onlover des vétements di
<sup>8</sup> Alloch Houtkontold: Soifonwoooorroinigung	<sup>8</sup> A product of point of the sector of the
<sup>3</sup> Nech Augenkentekt Gründliche Wesseren ühren Augenerster	Après le contact physique : nettoyage d'eau savoirneuse
Nach Augenkontakt Grundliche Wasserspulung, Augenarzt V	Après le contact des yeux. Inicage d'eau soigneuse, se présent
[Nach Verschlucken: Nicht Erbrechen Veranlassen, Arzt vorstei	Apres lavoir avaier. he pas lorcer de le rejeter, se presenter c'
Hinweise für den Arzt: sedierende bis narkotische Wirkung mo	Renseignements pourte medecini : ener d'apaisementjusqu'à
Jnach Verschlucken kontrollierte Magenspulung (siehe auch Si	Lavage d'estornac après l'avoir avaie (voir aussi fiche techniqui
5 - Yviaisnanmen zur Brandbekampfung	1º 5 Mesures de controles des incendies
Seeignete Löschmittel Schaum, Kohlendioxid, Löschpulver	Moyens d'extinction appropnes: Mousse, dioxyde de carbon, pi
<sup>13</sup> [Aus Sicherheitsgründen ungeeignete Löschmittel. Wasser	<sup>10</sup> Moyens d'extinction inapproprie pour raisons de sécunté: de le
"iBesondere Gefahrdung durch den Stoff, seine Verbrennungs	<sup>10</sup> Danger extraordinaire par la substance, par ses produits de cc
Besondere Schutzausrüstung für die Brandbekämpfung: -»-•	<sup>11</sup> Equipement de protection extraordinaire pour le contrôle des ii
	-
3→Mögliche Gefahren (durch das Lösungsmittel DK)	3Dangers qui peuvent apparaltre [par le dissolvant DK]
For Help, pressFl	German (Germany) -> French (France) 1 NUM f

Abb. 4.11: Alignment bei SDLX



Abb. 4.12: Alignment mit Trados WinAlign

Diese durch das System vorgenommene Aufteilung kann jedoch aus verschiedenen Gründen fehlerhaft sein. Wenn das System das Ende eines Satzes zum Beispiel durch das Vorhandensein eines Satzzeichens definiert, können fehlende Satzzeichen, wie zum Beispiel bei Überschriften oder Auflistungen, bzw. Punkte innerhalb eines Satzes, wie zum Beispiel bei Adressangaben, zu Fehlern führen. Außerdem wird in einer Übersetzung nicht zwangsläufig die Gesamtanzahl der Sätze beibehalten. Es kann durchaus sinnvoll sein, Sätze beim Übersetzen aufzuteilen, zusammenzufassen oder auch wegzulassen. So wird dem Käufer zum Beispiel in amerikanischen Handbüchern meist in der Einleitung zur Anschaffung des Produktes gratuliert, was in Deutschland hingegen unüblich ist, und deshalb oftmals in der Übersetzung weggelassen wird.

Deshalb muss der Übersetzer anschließend manuell das Ergebnis des automatisch erstellten Alignments korrigieren, bevor er es in ein Translation Memory importiert.

32

### 4.2.5 Maschinelle Übersetzung mit Translation-Memory-Systemen

Einige Translation-Memory-Systeme unterstützen die Einbindung von maschinellen Übersetzungssystemen in den Übersetzungsprozess oder bieten selbst eine Funktion zur Erstellung maschineller Übersetzungen an. Bei der Standardversion von *SDLX* können sowohl Systeme fremder Hersteller als auch das von *SDL* entwickelte System *AutoTrans* eingebunden werden. Bei den *SDLX*-Versionen für (Übersetzungs-)Unternehmen, *Professional* und *Elite*, ist *AutoTrans* im Lieferumfang enthalten.

#### 4.2.6 Statistiken und Berichte

Eine weitere Funktion vieler Translation-Memory-Systeme ist die Erstellung von Statistiken und Berichten, auch Analyse-Funktion genannt. Mithilfe dieser Funktion können Informationen über die Ausgangsdateien gewonnen werden, wie zum Beispiel die Anzahl der zu übersetzenden Segmente, Wörter und Zeichen, und wie viele Wiederholungen es innerhalb der Dateien gibt. Auch der aktuelle Stand bzw. Fortschritt des Übersetzungsprojektes kann ermittelt werden. Außerdem können die Dateien mit vorhandenen Übersetzungsspeichern verglichen werden und die Anzahl der Matches und deren Matchwerte aufgeschlüsselt werden. Anhand dieser Informationen kann unter anderem der Nutzen des Einsatzes vorhandener Übersetzungsspeicher abgeschätzt werden. Mit einigen Systemen, wie zum Beispiel *Trados* und *SDLX*, können auch so genannte Termkandidaten ermittelt und extrahiert werden, die dann im Vorfeld schon recherchiert und gegebenenfalls an die Übersetzer weitergegeben werden können. Die Analyse-Funktion spielt somit eine wichtige Rolle bei der Zeit- und Kostenplanung von Übersetzungsprojekten.

#### 4.2.7 Unterstützte Formate

Die aktuellen *Trados*-Versionen 6,5 bzw. 7 unterstützen getaggte Textformate wie zum Beispiel HTML, XML und SGML. *Trados* 7 bietet auch mehrere vordefinierte Tag-Einstellungsdateien, unter anderem für XLIFF (XML Localization Interchange File Format), ein Standard, der entwickelt wurde, um viele unterschiedlich zu lokalisierende Daten auf ein einheitliches Format abbilden zu können. Außerdem werden von *Trados* Formate von Desktop-Publishing-Programmen wie MIF *{Framemaker}*, IASCII (*Interleafund QuickSilver*) oder QSC (*QuarkXPress*) unterstützt. Die *Microsoft-Office-Frodukte Word, Excel* und *PowerPoint* können mit *Trados* ebenso bearbeitet werden wie *Microsoft-Windows-32*-Softwaredateien. Diese Softwaredateien lassen sich grob in zwei Formattypen unterteilen. Zum einen sind das Ressourcendateien, die als reine Textdateien den Quellcode der Elemente der Benutzeroberfläche enthalten. Diese müssen nach der Übersetzung wieder mit dem übrigen Quellcode zusammengeführt und kompiliert werden. Zum anderen sind es ausfthrbare Dateien, die den Programmcode in kompilierter Form enthalten, wie zum Beispiel EXE- oder DLL-Dateien.

Die Bandbreite der von *SDLX 2005* unterstützten Formate ist ähnlich. Unterstützt werden Code-Dateien (.txt, .c, .cpp, .h, .frm, .bas, .ctl, .eis), CopyFlow-Filter (.tag), TXT-, CSV- und REZ-Dateien, HTML-Dateien, Java-Properties-Dateien, MIF-Dateien, Ressourcendateien, RTF-Dateien (einschließlich .src), SGML-Dateien (einschließlich .dtd), WBF-Filter (.htm, .html), XML, ASP- und JSP-Dateien sowie *Word, Excel* und *PowerPoint* von *Microsoft Office*.

# 4.3 Lokalisierungstools

### 4.3.1 Allgemeines

Lokalisierungstools basieren auf der Translation-Memory-Technologie und sind speziell für die Lokalisierung von Softwareanwendungen entwickelt worden. Mit ihnen lassen sich die graphischen Benutzeroberflächen von Softwareanwendungen im Übersetzungseditor anzeigen und manipulieren. So hat der Übersetzer einerseits die Möglichkeit, die aktuelle Übersetzungseinheit in ihrem genauen Kontext zu sehen, und andererseits kann er die Größe und Lage der Elemente der Benutzeroberfläche der Länge der Übersetzungen anpassen.

Außerdem besitzen Lokalisierungstools umfangreichere Qualitätssicherungsfunktionen als Translation-Memory-Systeme, die genau auf die Prüfung von Softwareanwendungen zugeschnitten sind.

Zwei der den Markt anführenden kommerziell erhältlichen Lokalisierungstools sind die Systeme *Passolo* der Firma *Pass Engineering* und *Catalyst* der Firma *Alchemy*. Diese werden im folgenden Kapitel vorgestellt<sup>5</sup>.

#### 4.3.2 Passolo

#### 4.3.2.1 Allgemeines

Die aktuelle Version von *Passolo* ist die Version *5,0. Passolo* zeichnet sich besonders durch seine hohe Benutzerfreundlichkeit aus.

Mit *Passolo* können multilinguale Projekte bearbeitet werden. Im Projektfenster von *Passolo* werden die Quelldateien und die ihnen zugeordneten zielsprachlichen Versionen in unterschiedlicher Form dargestellt.

In der so genannten **Baumansicht** (vgl. Abbildung 4.13) erscheinen diese Elemente im oberen linken Teilfenster des Projektfensters. Im rechten oberen Teilfenster sind Textlisten dargestellt, die nach ihren Sprachversionen geordnet sind. Die erste bzw. oberste Textliste repräsentiert die Ausgangssprache, danach folgen die Zielsprachen in der Reihenfolge, in der

<sup>&</sup>lt;sup>5</sup> Für die Diplomarbeit verwendete Versionen: Passolo 5.0.004, Catalyst 5.03 (beides Demoversionen)

sie festgelegt wurden. Unterhalb dieser beiden Fenster lassen sich das **Ausgabefenster** und das **Eigenschaftsfenster** einblenden, in denen zusätzliche Informationen wie Meldungen der Software oder Ergebnisse einer Qualitätsprüfung angezeigt werden können.



Abb. 4.13: Das Projektfenster von Passolo in der Baumansicht
In den Ansichten **Quelldatei-Gruppierung** und **Sprach-Gruppierung** werden die einzelnen Elemente nicht tabellarisch aufgelistet, sondern durch Schaltflächen repräsentiert. Dabei werden bei der Ansicht **Quelldatei-Gruppierung** die Schaltflächen, die die zielsprachlichen Dateien repräsentieren, eingerückt dargestellt (vgl Abbildung 4.14).



Abb. 4.14: Das Projektfenster von Passolo mit Quelldatei-Gruppierung

In Abbildung 4.15 ist die Ansicht **Sprach-Gruppierung** dargestellt. In dieser Ansicht sind alle Elemente linksbündig angeordnet, und die Zielsprachen werden durch eine größere Schriftgröße hervorgehoben.



Abb. 4.15: Das Projektfenster von Passolo mit Sprach-Gruppierung

Alle zu übersetzenden Texte einer Ausgangsdatei werden in der so genannten Quelltextliste aufgeführt. Die Übersetzungsliste für die Zieldatei besteht aus einer Kopie der Quelltextliste und einer zusätzlichen Spalte für die übersetzten Texte. Nachdem alle Textlisten übersetzt worden sind, können die zielsprachlichen Dateien über den Menüpunkt **Zieldatei generieren** aus dem Menü **Textliste** erzeugt werden, indem eine Kopie der Quelldateien angelegt wird, in der alle Textressourcen entsprechend der Übersetzung geändert sind (vgl. Abbildung 4.16).



Abb. 4.16: Eckpunkte eines Lokalisierungsprozesses mit Passolo<sup>6</sup>

Bei einigen Dateiformaten, wie zum Beispiel ausführbaren Programmdateien oder XML-Dateien, können die übersetzten Ressourcen auch in die Quelldatei zurückgeschrieben werden, so dass alle Sprachversionen in einer einzigen Datei enthalten sind (vgl. Abbildung 4.17).



Abb. 4.17: Die übersetzten Texte werden in die Quelldatei zurückgeschrieben<sup>6</sup>

<sup>&</sup>lt;sup>6</sup> Abbildung erstellt auf der Grundlage der Online-Hilfe von Passolo 5.0

## 4.3.2.2 Grundlegende Aspekte bei der Übersetzung mit Passolo

Der Übersetzungsstatus der einzelnen Segmente wird bei *Passolo* durch unterschiedliche Farben repräsentiert. Rote Schrift repräsentiert unübersetzten Text, grüne Schrift automatisch übersetzten, ungeprüften Text. Blaue Schrift wird bei manuell übersetztem und ebenfalls noch nicht geprüftem Text verwendet und schwarzer Text repräsentiert übersetzte und geprüfte Übersetzungseinheiten. Der Status von Übersetzungseinheiten kann auch durch den Benutzer geändert oder zurückgenommen werden.

Die Bearbeitung der Texte erfolgt im Teilfenster Eigenschaften, welches sich unten rechts auf dem Bildschirm befindet.

Das Anpassen von Dialogfeldern und deren Elementen, wie zum Beispiel die Lage und die Größe von Schaltflächen und Labels, erfolgt im **Dialogeditor.** In diesem Editor werden die Elemente der zu lokalisierenden Software in der so genannten WYSIWYG<sup>7</sup>-Ansicht dargestellt (vgl. Abbildung 4.18).

<sup>&</sup>lt;sup>7</sup> Abkürzung für das Prinzip What You See Is What You Get (http://de.wikipedia.org/wiki/WYSIWYG, Zugriff: 07.07.2005)

	7 <b>%</b>			Ţ.		212
O Dialog						
E 22"About"	100 10					
I E 23	And all house					
1	CL	ne				
L E 25 "Managa Files"	CO	olumn a				
E 26 Copy/Repair	CP	age 🦯				
E 27 "Delete/Touc	CB	yte		1		
f- M 28 "Find In Files"	CN	ext Books a	the second se	,		
i- E 29 "Purchase Remade		21 11 1		0		
- E 30 "Go To"		Mand select	01			
E 31 "ChooseFile"	1	io To	Abbrechen			
1-[P] 33 "Tools"	-					
E 34" <tool>"</tool>						
- E 35 "Find"	1.	80 44				
- E 36 "Replace"	D#+ /#5		HXISM	m I		64 j 79   50 x 14 [[
; m 37	100	200	-			
E 38 "Compare Text File	Numm	er JID	Englisch (U5A)	Deutsch (Deutschland)	Kommentar	
IE 39 "Run"	EI Dialag	30 "Go To"				
IE 42 "Quick Open File"	643	0	Go To	GoTo		
! g] 43 "Document Class W	646	161	Extend selection	Extend selection		
! E 44 "Document Class Vv	647	1	Go To	Go To		
	648	2	Csncel	Abbrechen	alling grant and a second second	
I E 45 "Document Class Vo	640	271	Line	Line		
I E 45 "Document Class to E 46 "Document Class to	049	272	Column	Column		
E 45 "Document Class to    E 46 "Document Class to    M\ 47 "Tool Parameters"	650					
<ul> <li>E 45 "Document Class to</li> <li>E 46 "Document Class to</li> <li>M 47 "Tool Parameters"</li> <li>E 48 "Prompt"</li> </ul>	650 651	273	Page	Page		
E 45 "Document Class to     E 46 "Document Class to     M/ 47 "Tool Parameters"     E 48 "Prompt"     E 49 "Prmt"	650 651 652	273 274	Page	Page		
I E 45 "Document Class to E 46 "Document Class to M1 47 "Tool Parameters" E 48 "Prompt" E 49 "Prmt" = E3 Rtl "Gn Tŋ Artrip^^	650 651 652 653	273 274 275	Page Byte Next Bookmark	Page Byte ⊈t.≵⊊uokmark		
LE 45 "Document Class to E 46 "Document Class to M 47 "Tool Parameters" E 48 "Prompt" E 49 "Prmt" — E3 Rtl "Gn Tn Artrip*	650 651 652 653	273 274 275	Page Byte Next Bookmark	Page Byte 12.1. Quokmark	usa - Schaltijäche 2	Sin Dialog Di
LE 45 'Document Class to 	650 651 652 653	273 274 275	Page Byte Next Bookmark	Page Byte 11.*.Guokmark * Denset tunger	est - Schalt liche P	tin Dialog 30 ✓ 12   0
LE 45 'Document Class to 	650 651 652 653 PiNfung j	273 274 275 fä Sucher	Page Byte Next Bookmark	Page Byte 17:X.Guokmark Chersetzunguster Text Eigenschaft	en   Info       @	Ha Dialog 10 ✓ 15   ②
E 4 5 'Document Class to	650 651 652 653 PiNfung j	273 274 275 fã Suche	Page Byte Next Bookmark	Page Byte 11: X Suokmark K Obersetzungst Text Eigenschaft	en Info   1 @	Hn Dfalog 20 ✔ 1월   ②
LE 45 'Document Class to 	650 651 652 653 PiNfung j	273 274 275 <i>fä</i> Sucher	Page Byte Next Bookmark	Page Byte 17: X. Quokmark × Diterset zungeb Text pageschaft	en Info   1   Ф	In Dialog 20
LE 45 'Document Class to 	650 651 652 653 PiNfung j	273 274 275 fä Sucher	Page Byte Next Bookmark	Page Byte 1:X. Juokmark × (Diarsectum) Hechagen	en Info   1 👁	Hn Didoog 20 ▼ 15   O
E 45 'Document Class to	650 651 652 653 PiNfung j	273 274 275 fä Sucher	Page <u>Byte</u> <u>Next Bookmark</u> n   @ Aufgaben   ⊡→⊇ Nattaung	Page Byte 1: X. Juokmark Charse transe text Eigenschaft Heekunit	en Irfo     @	In Dielog 20 ✓ 12 O
LE 45 'Document Class to 	649 650 651 652 653 PiNfung j	273 274 275 fä Sucher	Page Byte Next Bookmark	Page Byte 1: * Zuokmark × Dibersetzungeb teschlagen Herkunft Cancel Abbrechen	en Irfo   1 @	In Dialog 20 ✓ 15 O
LE 45 'Document Class to 	649 650 651 652 653 PiNfung j	273 274 275 fä Sucher	Page Byte Next Bookmark	Page Byte 11:X: Zuokmark X: Dibersetztmin de Text Digenschaft Digenschaft Digenschaft Cancel Abbrechen	en Info     @	10 Dialog 30 ✓ 12 0 1
E 4 5 'Document Class to	649 650 651 652 653 PiNfung j	273 274 275 fä Sucher	Page Byte Next Bookmark	Page Byte 12: X. Suokmark Concection Herkunft Cancel Abbrechen	en Irfo     @	±n Dialog 10 ✓ 12 0

Abb. 4.18: Die Arbeitsumgebung bei *Passolo* mit dem Teilfenster **Eigenschaften** (1) und dem **Dialogeditor** (2)

Passolo besitzt eine Funktion Automatische Übersetzung, welche Übersetzungslisten und Glossare, die zuvor in das Projekt eingegliedert wurden, als Quelle verwendet. Diese lässt sich auf einzelne Übersetzungseinheiten oder auch auf die gesamte Textliste anwenden und über die Menüpunkte Text automatisch übersetzen bzw. Automatisches Übersetzen... im Menü Übersetzung auswählen.

Das Teilfenster unten links auf dem Bildschirm in den Abbildungen 4.19 und 4.20 heißt Ausgabefenster. In der Registerkarte Übersetzung dieses Teilfensters werden die Ergebnisse der Fuzzy-Match-Suche und der Konkordanzsuche angezeigt.



Abb. 4.19: Das Ausgabefenster mit Ergebnissen der Fuzzy-Match-Suche

	98	<b>B</b>	ę 🔂 🖄 🛍				
El 86 "Document Classe6, El 87 "Preferences" El 83 "Syntax" El 89 "Syntax" El 90"TextPad" El 91"Spelling" •El 92 "Cremte Dictionary" M 93 "Edt Dictionary" El 94 "Create Dictionary El 95 "Language" El 96 "Keyboard" El 96 "Keyboard"		nilbar r name:	tfe_	neg ntiud I ep f			
El 20509 "Commands" El 20511 "New Toolbar" El 20513 "Toolbars" El 23150	  2+ +0	## <u>##</u>	HIIB				117 , 24 50 x 14
El 30721 "New"	Nummer	ID	Englisch (USA)	Deutsch (Deutschla	nd) Kommentar		
EI 30722	P Dialag 2	0511 "New "	Toolbar"				
<b>E</b> 30723	1358	0	New Toolbar	New Toolbar			
) 5tring Table	1360	-1	Toolber name:	Toolbar name:			
Contraction Tabelle	1362	1	QK	QK			
i 🔁 Cursor	1363	2	Gancel	Gancel			
Icon	1364	57670	Help	Help			
Manifest ToolBar	✓ Phillung	M Suchen	😨 Aulgaben		X	Übersetzungs Text Elgens	text 1963 - Schaltflache 2 in Dr chaften   Info   ↓ 🚺 👁 🗸 15
Jocarca		1/2		lust an	Nacisci en		
onkordanz  Quele 00.% Cancel		Abbrech	ung	TevtRad, Entry 33312 in M	en0.20	&Cancel	
00 % Cancer Some of your doc been modified. So none of them, cho save, or cancel th	uments have ave all of them, iose which to is command?	Einige De geänder speicher ausgewä	okumente wurden t. Möchten Sie alle n, keins speichern, ihlte speichern oder	TextPad, Static in Dialog 7:	, il il	Rearce)	

Abb. 4.20: Das Ausgabefenster mit Ergebnissen der Konkordanzsuche

Die Tastenkombinationen für Zugriffstasten können entweder manuell durch ein Und-Zeichen (&) vor dem jeweiligen Buchstaben des Wortes vergeben oder auch automatisch durch *Passolo* vergeben werden. Dies kann entweder durch drücken der Taste F9 geschehen, oder über den Befehl **Zugriffstaste einfügen** im Menü **Übersetzung.** Dadurch wird sichergestellt, dass die Zugriffstasten nicht doppelt vergeben werden können.

Zu jeder Übersetzung kann in einer eigens dafür vorgesehenen Spalte ein Kommentar abgelegt werden.

Bei *Passolo* gibt es zwei Möglichkeiten, einmal angefertigte Übersetzungen für die spätere Wiederverwendung nutzbar zu machen. Sie können entweder in ein Projektglossar eingefügt werden, oder direkt in ein Translation Memory eingetragen werden, welches vorher für das Hinzufügen von Segmenten aktiviert wurde. Dazu muss der Befehl **Übersetzung speichern** bzw. der Befehl **An ein Übersetzungs-Add-In senden** im Menü **Übersetzung** ausgewählt werden.

43

Als Exportformate stehen bei *Passolo* eine Passo/o-Übersetzungsdatei, das Textformat, ein Posso/o-Glossar oder durch Add-Ins unterstützte Formate zur Verfügung. Gültige Importformate sind eine Passo/o-Übersetzungsdatei, das Textformat, ein *Passolo*-Übersetzungspaket (beim Aufteilen von Projekten auf mehrere Übersetzer) oder durch Add-Ins unterstützte Formate

*Passolo* verfügt über eine große Anzahl an Prüffunktionen, wie zum Beispiel die Suche nach fehlenden Übersetzungen, fehlenden oder doppelt vergebenen Zugriffstasten oder abgeschnittenem Text, welche die Qualitätssicherung erleichtern. Das Teilfenster **Prüffunktionen** im Fenster **Optionen** ist in Abbildung 4.21 abgebildet.

Optionen	<u>? × </u>
öV       Ansicht         fnn       Symbolleisten         Image: Construction of the symbol of th	<ul> <li>Prüffunktionen</li> <li>Diese Prüfungen werden an übersetzten Texten durchgeführt:</li> <li>I Alle Texte <ul> <li>Unterschiedliche Anzahl an Tabulatoren.</li> <li>In der Übersetzung fehlt die Zugriffstaste ('8t'),</li> <li>Die Übersetzung benötigt keine Zugriffstaste.</li> <li>Fehlende Übersetzung.</li> <li>Ein leerer Quelltext hat eine nicht-leere Übersetzung.</li> <li>Unterschiedliche Anzahl von Leerzeichen am Anfang oder Ende des Te;</li> <li>Zugriffstaste außerhalb des erlaubten Bereiches.</li> <li>Text ist zu langD</li> <li>Rechtschreibungs-Fehler</li> <li>Menüs</li> <li>Doppelte Zugriffstaste im Menü.</li> </ul> </li> <li>Dialoge <ul> <li>Das Steuerelement ist nicht groß genug für den enthaltenen Text.</li> <li>Ignoriere Größen-Fehler, die bereits in der Quelldatei auftreten.</li> <li>Das Steuerelement überlappt ein anderes.</li> <li>Das Steuerelement liegt nicht innerhalb der Dialogbegrenzung.</li> <li>Doppelte Zugriffstaste definiert.</li> </ul> </li> </ul>
	OK   Abbrechen Hilfe

Abb. 4.21: Prüffunktionen bei Passolo

#### 4.3.2.3 Weitere Leistungsmerkmale von Passolo

Eine weitere Funktion von *Passolo* ist die Erstellung von Statistiken und Berichten, welche in ihrer Ausführlichkeit je nach Programmversion (*Standard Edition* oder *Professional Edition*) variieren. Wie der Abbildung 4.22 zu entnehmen ist, kann mit der *Standard-Version* die Anzahl der zu übersetzenden Texte, die Anzahl der zu übersetzenden Wörter, der prozentuale Anteil der bereits übersetzen Texte sowie eine prozentuale Angabe bezüglich der Textlängenänderung berechnet werden.

Information		BEI
Textliste:	TextPad: Deutsch (Deutschland)	
Anzahl der Texte:	2121	
Wortanzahl:	7139	
Bereits übersetzt:	4%	
Änderung der Textlängen:	127 %	
	СК	

Abb. 4.22: Eine Statistik bei der Standard-Version von Passolo

Der Benutzer der *Professional-Version* kann darüber hinaus noch weitaus detailliertere bzw. die in Abbildung 4.23 dargestellten statistischen Auswertungen abrufen.

Beschreibung	Texte	Wörter	Buchstaben	Zeichen
Gesamtzahl der Texte	357	2.305	12.228	15.935
Gesamtzahl der zu übersetzenden Texte	357	2.305	12.228	15.935
Anzahl der unübersetzten Texte	296 (82%)	2.244 (97%)	11.948(97%)	15.614 (97%)
Anzahl der unübersetzten, wiederholten Texte	61 (17%)	61 (2%)	280 (2%)	321 (2%)
Anzahl der zu prüfenden Texte	0 (0%)	0 (0%)	0 (0%)	0 (0%)
Anzahl der übersetzten und validierten Texte	0 (0%)	0 (0%)	0 (0%)	0 (0%)
Änderungen der Textlängen in der Übersetzung	0 (0%)	0 (0%)	0 (0%)	0 (0%)

Abb. 4.23: Eine Statistik bei Professional-Version von Passolo

Außerdem können mit *Passolo* Übersetzungen simuliert werden. Dies dient zur Prüfung der generellen Übersetzbarkeit einer Anwendung (im Sinne der technischen Durchführbarkeit) und zur Prüfung der Stabilität einer Anwendung, welche zum Beispiel durch die Längenänderung von Texten eingeschränkt werden kann. Des Weiteren kann mit einer simulierten Übersetzung im Vorfeld geprüft werden, ob sprachspezifische Zeichen angezeigt werden können.

#### 4.3.2.4 Schnittstellen für Glossare und Translation-Memory-Systeme

*Passolo* unterstützt Glossare im eigenen Glossarformat, welches tabulatorgetrennt ist und die Dateinamenserweiterung **.glo** besitzt, sowie tabulatorgetrennte TXT-Dateien. Auf der Internetseite von *Passolo<sup>8</sup>* ist ein kostenloses Editorprogramm erhältlich, mit dem man *Microsoft-Glossare* in *Passolo-GXossarQ* konvertieren kann.

Add-Ins für Translation-Memory-Systeme sind derzeit nur für die *Trados Workbench* und *Star Transit* erhältlich, laut der Internetseite von *Passolo* ist die Unterstützung anderer Systeme jedoch für die Zukunft vorgesehen.

#### 4.3.2.5 Von Passolo unterstützte Dateiformate

Bei *Passolo* wird die Unterstützung von Dateiformaten mittels Add-Ins realisiert. Für einige Dateiformate sind dafür in der kommerziell erhältlichen Version des Programms bereits Add-Ins vorhanden. Für andere Dateiformate können Benutzer mit guten Programmierkenntnissen auch Add-Ins entwickeln und einbinden.

Standardmäßig sind bei *Passolo* Add-Ins für ^7«öfow\*-32-Bit-Binärdateien wie zum Beispiel EXE-, DLL- und OCX-Dateien sowie Programme für *Windows CE* oder *Pocket PC*, für J^Wows-lö-Bit-Binärdateien wie zum Beispiel EXE-, DLL- und SYS-Dateien und für *Windows-RC-Dateien* wie zum Beispiel RC2- und DLG-Dateien verfügbar. Außerdem enthält die *Standard*-Version von *Passolo* Add-Ins für Visual-BASIC-Binärdateien, XML-Dateien, Textdateien und Mcr0.ro/Mnstaller-Dateien. Mit dem Add-In für XML-Dateien können auch ID-basierte XML-Dokumente wie zum Beispiel XLIFF-Dateien bearbeitet werden. Dafür muss zuvor die Struktur der XML-Datei definiert werden.

Optional können Add-Ins für .NET-Programme von *Microsoft*, für Borland-Delphibzw. C++-Builder-Programme und für Java-Programme erworben werden.

Außerdem stellt *Passolo* Makros für die Bearbeitung von *Windows*-Konfigurationsdateien, von Java-Property-Resource-Dateien und für ^cce^-Datenbanken zur Verfügung. Mithilfe einer integrierten Entwicklungsumgebung kann der Benutzer weitere Parser für eigene Datenformate entwickeln.

<sup>&</sup>lt;sup>8</sup> http://www.passolo.com/support/downloadmacros.stmSMSGlos, Zugriff: 06.07.2005

## 4.3.3 Catalyst

#### 4.3.3.1 Allgemeines

Die aktuelle Version von *Catalyst* ist die Version 6.0. Obwohl die grundsätzliche Funktionsweise der beiden Systeme *Passolo* und *Catalyst* ähnlich ist (Extraktion des zu übersetzenden Textes und Erstellung von sprachspezifischen Textlisten, die nach der Übersetzung in die Zieldateien eingefügt werden), können mit *Catalyst* keine multilingualen Projektdateien angelegt werden. Für jede Sprachrichtung eines Lokalisierungsprojektes muss eine eigene Projektdatei erstellt werden.<sup>9</sup>

Wie die nachfolgenden Abbildungen 4.24 und 4.25 zeigen, ähnelt der Aufbau der Benutzeroberfläche von *Catalyst* dem von *Passolo*. Auch bei *Catalyst* kann eine Baumstruktur aller in das Projekt eingebundenen Dateien angezeigt werden und die Übersetzungseinheiten der aktuell geöffneten Datei können entweder in tabellarischer Form oder in einer WYSIWYG-Ansicht dargestellt werden. Der untere Teil des Bildschirms enthält, wie bei *Passolo*, zwei Fenster, eines mit projektspezifischen Meldungen und Informationen, das andere dient zur Bearbeitung der Übersetzungseinheiten.

Ein Unterschied zu *Passolo* besteht in der Modularisierung von bestimmten Funktionen des Systems, auf die über das Teilfenster **Experts** am linken Bildschirmrand zugegriffen werden kann.

<sup>&</sup>lt;sup>9</sup> Bezugsquellen: Online-Hilfe Catalyst 5.03 und Support-Anfrage bei SAMEngineering GmbH

Alchemy Catalyst 5.03 Evaluration Copy - Ltcr	ntpüd.ttk I				
					2012
🖞 😂 🔜 🔄 🔢 🔛 🚅  English (United	Kingdom)	] Lieman (Liemany)	RFFFE		
orto - x Navigator - x	Start Pag	e String Mode WYSIWYG			
opens V 🖸 🕼 (no filter) 🔹	String ID	Translated		Original	
	0	<ul> <li>Dateien verwalten</li> </ul>		Manage Files	
- 17	65535	Working tolder:		Working folder.	
struate - 18	65535	🦈 &Dateien:		&Files.	
	151	&Copy	$\neg$	&Copy	
- 20	152	&Delete		&Delete	
	153	&Rename	<u> </u>	&Rename	i. ii
	154	&Touch	1968	&Touch	
20620	155	&Browse		&Browse	1
E-B Dialog	2	C&lose		C&lose	
- 22 - About					
- 🚍 🐨 25 - Dateien ve					
26 - Conv/Bery					
27 - Delete/To					
28 - Eind In File					
- 29 - Purchase F					
21 - Choose Elk					
33 - Choose Fax					
34-(1005					
Ja-Find					
<u>+</u>	í 🛛				
textpad tik					
	J				
Objeth		Description	× S Fromes		
Objeib			4 CLIOWSE.		Lengh 10 10
					MaxLen, I
4			&Browse.	-	
/					
				Naelized Datails	
				ADurchsuchon	-
			Bi Bruss		
		•	BIOWSE.		
ral A My Todo List A Replica A Validate Expe	rt A Leverage	Update Euperl A Find & Replace^	IMINI		*
			Produced and the product of the prod		
					NUM

Abb. 4.24: Die Arbeitsumgebung bei *Catalyst* mit dem Teilfenster **Experts** (1), Baumstruktur der Projektdateien (2), Liste der Übersetzungseinheiten (3), Teilfenster für projektspezifische Informationen (4) und Teilfenster zur Bearbeitung der Übersetzungseinheiten (5)

HAlchemy Catalyst 5.03 Evaluation Copy - [tcKtpad.ttk]	
B 2 B 1 ~& [[English(UnitedKingdüm] _J[Gamai(Geimanrf ⊻ □□□□ E e f &	
Shortci + jt Mavigator ^ 3 Start Page   String Mode   WS   WG	
Experts V b luc. filter	
Dateien verwaiten	
	Tor.
20 P Delete Rename Jouch   Browse   Close	
Leverage b-1 67	-
	<u> 7</u>
	S
Lipdate 22-About - 1 23-	
	X
Praudo -3 26-Copy/Ren	Н
-3 27-Delele/Toi	m
-1 28-Find In File -3 29-Purchase F	
Validate -1. 30-GoTo	œ,
	Q
Compare -3 34- <toob< td=""><td>2</td></toob<>	2
-3 35-Find	
Reso testpad.ttk	
* File Object Description & &Copy	Propeilj   T. 10   ••
	Length 8 8 MaxLe
8.Copy	x_lele"!
8 − 0toriginal Ecolated	
A Akopieren	
1 1 \ Geneial /\ H'j Todo List A Replica ^Validale Eupert \ Leveiage/Updale Eupert \ Find t Replace, 🖹 🕅 🕨 🕅 🔛 🔛 🦉 🧐 😿	*
Ready	NUM

Abb. 4.25: Die WYSIWYG-Ansicht bei Catalyst

#### 4.3.3.2 Grundlegende Aspekte der Übersetzung mit Catalyst

Der Übersetzungsstatus der Segmente wird bei *Catalyst* durch Symbole wiedergegeben. Ein Auge repräsentiert dabei übersetzte, nicht geprüfte Übersetzungseinheiten und ein Haken übersetzte und geprüfte Übersetzungseinheiten.

Um auf bereits vorhandene Übersetzungen zurückzugreifen, können die Funktionen Leverage oder Update genutzt werden. Mit dem Leverage Expert wird eine Vorübersetzung innerhalb der gesamten Projektdatei durchgeführt, mit dem Update Expert ist dagegen eine Vorübersetzung in einzelnen Dateien möglich. Die Suche nach Matches kann innerhalb einer Cato/j^^-Projektdatei (TTK-Datei), eines mit *Trados* angefertigten Übersetzungsspeichers (TMW-Datei), einer TMX-Datei oder einer tabulatorgetrennten Textdatei stattfinden.

50

Der Export kann bei *Catalyst* als TMX-Datei, als Glossar (getaggtes oder tabulatorgetrenntes Format), als *Workbench-Datei* oder als *Multiterm-Datei* erfolgen.

Die Ergebnisse der Suche in Glossaren werden während der Übersetzung automatisch in dem Teilfenster **Translator's Toolbar** unten links auf dem Bildschirm angezeigt.

Auch *Catalyst* verfügt über eine große Anzahl an Qualitätssicherungsfunktionen, welche speziell auf die Lokalisierung von Software abgestimmt sind. Die Prüffunktionen werden mit dem so genannten **Validate Expert** ausgeführt. Statistiken können mit dem so genannten **Report Expert** angezeigt werden, und mit dem so genannten **Pseudo Expert** können Übersetzungen simuliert werden.

Eine Funktion von *Catalyst*, die in *Passolo* nicht enthalten ist, stellt der so genannte **Compare Expert** dar, mit dem die aktuell zu lokalisierende Anwendung mit einer zweiten Anwendung, zum Beispiel einer Vorgängerversion der aktuell zu lokalisierenden Anwendung, verglichen werden kann. Somit kann bei Folgeprojekten schnell und unkompliziert ermittelt werden, welche Ressourcen hinzugefügt, entfernt oder verändert wurden, was die Zeit- und Kostenplanung von Lokalisierungsprojekten vereinfacht.

#### 4.3.3.3 Schnittstellen für Glossare und Translation-Memory-Systeme

In ein *Catalyst-Frojekt* können bis zu drei Glossare eingebunden werden. Ein Glossar davon kann die Übersetzungen enthalten, die während des aktuellen Projektes angefertigt wurden. Die beiden anderen Glossare können früher angefertigte Übersetzungen enthalten, oder auch mit anderen Anwendungen erstellt worden sein. Um ein solches Glossar einbinden zu können, muss es als tabulatorgetrennte Textdatei oder im TMX-Format vorliegen. Optional kann eine Komponente für den Datenaustausch mit *Trados* erworben werden.

#### 4.3.3.4 Von Catalyst unterstützte Dateiformate

*Catalyst* unterstützt alle M/cro\*q//'-Desktop-Entwicklungsplattformen (9x, NT, 2000, XP, Win32, RC, RESX, .NET Binaries, Visual Basic), die Mobile-Computing-Plattformen *Windows CE, Symbian, EPOC,* die Java-Plattformen J2EE, J2SE, J2ME, die Contentdateien *Microsoft Excel* und XML (einschließlich die abgeleiteten ASP.NET, ASP, JSP und XSL) sowie als Datenquellen alle M/craso/z'-DB-Technologien, *Oracle 8/9/10* und *IBMDB2*.

Optional können zwei Komponenten erworben werden, die eine visuelle Lokalisierungsumgebung für .NET-Anwendungen bzw. für den XLIFF-Standard ermöglichen, sowie eine Komponente zur Übersetzung von Datenbankinhalten.

# 5 Das Lokalisierungstool L10N

## 5.1 Besonderheiten von VisualWorks

Die Smalltalk-Entwicklungsumgebung *VisualWorks* besteht aus drei Komponenten. Das sind die graphische Benutzeroberfläche mit Werkzeugen zur Programmentwicklung, wie zum Beispiel dem so genannten **Arbeitsbereich (Workspace)**, ein Werkzeug, mit dem Smalltalk-Programme eingegeben und getestet werden können, die virtuelle Maschine für die Ausführung der Smalltalk-Programme sowie der Objektspeicher (auch Image oder Klassenbibliothek genannt), welcher alle Objekte einschließlich der Klassen und deren Methoden enthält, die von der virtuellen Maschine bearbeitet werden.

Der System-Browser, mit dem die Klassenbibliothek durchsucht werden kann, enthält fünf Teilfenster, jeweils eins für die Klassenkategorien und deren Klassen, die Methodenprotokolle der ausgewählten Klasse, die Methoden des ausgewählten Methodenprotokolls und für den Smalltalk-Code der jeweils ausgewählten Klasse oder Methode.

Um *VisualWorks* betreiben zu können, werden vier Dateien benötigt. Das sind eine IM-Datei, welche den Objektspeicher enthält, eine CHA-Datei, in der alle Änderungen am Image protokolliert werden, eine SOU-Datei, in der alle Methoden enthalten sind, die in der mitgelieferten Original-Image-Datei enthalten sind, und die Datei, die das ausführbare Programm der virtuellen Maschine enthält (unter Windows eine EXE-Datei). Wie bereits in Kapitel 3.6 beschrieben, werden bei *VisualWorks* alle übersetzbaren Texte einer Anwendung in so genannten **Locales** abgelegt. Diese liegen als LBL-Dateien und als IDX-Dateien vor.

## 5.2 Allgemeines zu L10N

Die derzeit erhältlichen kommerziellen Lokalisierungstools sind vor allem für die Lokalisierung von *Windows-Programmen* konzipiert. Um auch mit *VisualWorks* erstellte Anwendungen einfach lokalisieren zu können, wurde von der Firma *Georg Heeg eK* das Lokalisierungstool *LION* entwickelt.

LION läuft auf allen Plattformen, auf denen auch VisualWorks läuft. Das sind namentlich Windows (95/98/ME/NT/2000), PowerMac, Intel Linux, AIX, SGI Irix, Compaq UNIX, HP-UX, und Solaris.

Im Unterschied zu den bisher genannten Lokalisierungstools und Translation-Memory-Systemen wird für das Lokalisierungsprojekt keine eigenständige Datei angelegt, sondern die im Übersetzungsspeicher abgelegten Daten werden in einer *VisualWorks-Image-Datei* abgespeichert. Die im Translation Memory gespeicherten Daten können durch einen Export in einer Exportdatei, zum Beispiel im TMX-Format, abgelegt werden.

Der gesamte übersetzbare Text einer Anwendung ist die Gesamtheit aller Texte, die vom Benutzer gesehen werden. In *VisualWorks* werden diese Texte als so genannte Benutzernachrichten bezeichnet. Die Untergliederung in Übersetzungseinheiten erfolgt nach Benutzernachrichten. Dabei stellt je eine Benutzernachricht eine Übersetzungseinheit dar.

Im Folgenden wird die Benutzeroberfläche von *LION* beschrieben und eine Evaluierung des Tools vorgenommen. Um diesen Teil der Diplomarbeit als Informationsquelle für zukünftige Lokalisierungsprojekte mit *LION* auch für nicht-deutschsprachiges Publikum zugänglich zu machen, befindet sich im Anhang der Arbeit eine Übersetzung dieses Kapitels in die englische Sprache.

## 5.3 Elemente der Benutzeroberfläche

Das Lokalisierungstool *LION* ist in der Programmierumgebung von *VisualWorks* 7.3.1 implementiert (vgl. Abbildung 5.1). Es lässt sich öffnen, indem aus der Symbolleiste das erste Symbol von rechts, ein blaues Buch, oder aus dem Menü **Window** der Menüpunkt **1.** [FUI] Localizer ausgewählt wird.



Abb. 5.1: Das Startfenster von VisualWorks mit der Schaltfläche für das Lokalisierungstool

Das Lokalisierungstool öffnet sich in einem neuen Fenster, welches in Abbildung 5.2 dargestellt ist.

🗟 Localizer					_ 8 ×
File Import Export					
Source Language Specification:	j C_none	Method containing selected UserMessage	UserMessages inthis Method:		
Target Language Specification:	de 💌	windowSpec in UI.SimpIsListEc)itor cl	Add Add Change Change		3
Katalog'   0.7% (41548)	ubels 📩	formation and a second se	Delete: Delete		
Abort: Abort AboutVisualWorks: About Visual V	Vorks	V•iindowSpec			4
Accept: Accept Accessors. Accessors Add: Add add add addButton: addButton AddNamedFont: Add New Named AddSubCarvas: AddSubCarvas AdvancedUtilities: Advanced AdvancedUtilities: Advanced Utilit AdvancedUtilities: Advanced Utilit AdvancedUtilities: Advanced Utilit AdvancedUtilities: Advanced Utilit AdvancedUtilities: Advanced Utilit AdvancedUtilities: Advanced Utilit AdvancedUtilities: Allow Command-Line AllowFarceLoading: Allow Command-Line AllowFattings: Allow Command-Line AllowFattings: Allow Command-Line AllowFattings: Allow Command-Line AllowFattings: Allow Command-Line AllowFattings: Allow Command-Line	Font es ubtle and wizari loading d-Line Expressi File-ins (-filein, I hand-Line Parce he Settings (-sei	UPainter new openOnClass: seir <resource: #canvas=""> <!--##UI.FullSpec}<br-->Window: #(#(UI.WindowSpec) #iabel: #(#(Kernel.UserMessage) Default-String J Add Translations: Translator-31. Juli 2005 Translator-31. Juli 2005 Translator-31. Juli 2005</resource:>	19 55:00; Hinzufügen 19:51:37; Hinzufügen 19:51:15; Hinzufügen 19:43:39; Hinzufügen		-
AmpHelp &Help		Existing Translations	Numberto show: 20	Q Matching: 70	* %
AnErrorOccurredInPageDefinition AnErrorOccurredInSetting: An err Application: Application ApplicationDevelopment Applica ApplicationDevelopmentDesc: Tra ApplicationServer: Application Se ApplicationServerDesc VisualWo Beopository: a Repository	1: An error has i or has occurred tion Developmet. rols and framewi rver rks Web Applic	98.0%Hinzufügen98.0%Hinzufuegen96.0%Addieren73.0%^Hinzufügen71.0%Hinzufügen*71.0%Hinzufügen:	(Add translated in labeis and 98 oth (fild translated in labeis and 0 othe (Add translated in Eitcei2+B and 2 (&Add translated in menus and 0 ot (Add> translated in WindowsServer (Add: translated in WindowsXP and	er catalogs) er catalogs) other catalogs) her catalogs) 2eH and 1 other catalog) 0 other catalogs)	-
AssignNamedFont: Assign To Nam	ned Font	YourTranslation			
AutoInstallWhenDefining: Auto-in AutoInstallWbenOpening: Auto-in	stall when defini stall when open	Hinzufügen		AcceptTran	Islation

Abb. 5.2: Das Lokalisierungstool L10N

In der Menüleiste befinden sich die drei Menüpunkte File, Import und Export. Über das Menü File werden die wesentlichen Funktionen erreicht, die für den Übersetzungsprozess benötigt werden. Das sind die Befehle Accept Translation (Strg+S), Read System und Write and Compile Catalogs (Umschalt+Strg+S).

Mit dem Befehl **Read System** (dieser kann alternativ auch im Dialogfeld **Settings** über die Schaltfläche **Edit...** aufgerufen werden) werden alle Methoden der zu übersetzenden Anwendung nach Konstrukten durchsucht, in denen Benutzernachrichten verwendet werden. Die Benutzernachrichten werden dann im Teilfenster **UserMessages** aufgelistet.

Wird der Befehl Accept Translation ausgewählt, wird die eingegebene Übersetzung bestätigt und im systemeigenen Translation Memory gespeichert.

Mit dem Befehl Write and Compile Catalogs werden die Übersetzungen in die LBLund die IDX-Dateien gespeichert, so dass sie auch in anderen F/swa/^Forfo-Installationen verwendet werden können.

Außerdem enthält das Menü File auch noch die Menüpunkte Options, Original Language, Empty TM und TM Stats.

Wählt der Benutzer den Menüpunkt **Options** aus, öffnet sich das Dialogfeld **Settings**, in dem die Einstellungen für das Lokalisierungstool vorgenommen werden können (vgl. Abbildung 5.3). Es enthält die Kontrollkästchen **Remove translated entries from the UserMessage list** (entfernen der übersetzten Einträge aus der Benutzernachrichten-Liste), **Skip translated entries in the UserMessage list** (übersetzte Einträge in der Benutzernachrichten-Liste überspringen), **Include translation from Cnone when 'en' is the source** (Übersetzungseinheiten mit der Ausgangssprache Cnone mit berücksichtigen, wenn Englisch als Ausgangssprache für die Lokalisierung gewählt wurde) sowie **Use CaseSensitive comparison for existing translations** (Groß- und Kleinschreibung beachten). Des Weiteren befInden sich in diesem Dialogfeld die drei Auswahllisten **Number of matches shown** (Anzahl der angezeigten Matches) mit den Optionen **Show all** (alle anzeigen), 1, 3, **10**, 25, 50, **100** sowie **Source directory** (Verzeichnis des Ausgangstextes) und **Target directory** (Verzeichnis des Zieltextes).



Abb. 5.3: Das Dialogfeld Settings

Im Dialogfeld **Penalties** (vgl. Abbildung 5.4), erreichbar über die Baumansicht im linken Teilfenster des Dialogfeldes **Settings**, lassen sich prozentuale Abzüge für Matchwerte gefundener Übersetzungsvorschläge einstellen. Abzüge sind in der Spanne von 0% bis 100% möglich, und diese können vergeben werden, wenn der ausgangssprachliche Text eine andere Länderkennung besitzt als der Ausgangstext des gefundenen Matches, wenn der zielsprachliche Text anders als der Zieltext des gefundenen Matches keine Länderkennung besitzt oder wenn der zielsprachliche Text eine andere Länderkennung besitzt als der Zieltext des gefundenen Matches keine Länderkennung besitzt als der Zieltext des gefundenen Matches keine Länderkennung besitzt als der Zieltext des gefundenen Matches keine Länderkennung besitzt als der Zieltext des gefundenen Matches keine Länderkennung besitzt als der Zieltext des gefundenen Matches.

v3>Settings		
Settings         Image: Properties state         Image: Properties state         Image: Properties state         Image: Properties state         Image: Properties Publishing Polic	-S Penalties Source Language other country: Target language neutral country: Target language other country:	2% 2% A%
	ncel	Help

Abb. 5.4: Das Dialogfeld Penalties

Über den Menüpunkt **Original Language** kann ausgewählt werden, ob die Benutzeroberfläche einer Anwendung in der Ausgangssprache oder in der Zielsprache dargestellt wird, und über den Menüpunkt **Empty TM** kann das systemeigene Translation Memory geleert werden. Über den Menüpunkt **TM Stats** kann angezeigt werden, wie viele Einträge sich zur Zeit im systemeigenen Translation Memory befinden (vgl. Abbildung 5.5).

*	×
617023 Entries are	currently in the TM
0	K

Abb. 5.5: Translation-Memory-Statistik

Das Menü Import enthält die Menüpunkte Import Catalog (Katalog importieren), Import CSV-Catalog (Katalog im CSV-Format importieren), Import CSV-Catalog Folder (Ordner mit Katalogen im CSV-Format importieren), Import TM (Translation Memory importieren), Import TMX (TMX-Datei importieren) und Import TBX (TBX-Datei importieren). Über diese Importfunktionen können Glossare, die als LBL-Datei oder als CSV-Datei vorliegen, systemeigene Translation-Memory-Dateien (Formate: Translation Memory (.tm), Smalltalk File-outs (.st), Smalltalk Parcel Sources (.pst), Smalltalk Change-Dateien (.cha), Smalltalk Parcels (.pcl), Workspaces (.ws), Textdateien) sowie TMX-Dateien (Version 1.1) und TBX-Dateien importiert werden.

Im Menü **Export** befinden sich die Menüpunkte **Export TM** (als Translation Memory exportieren), **Export TMX** (als TMX-Datei exportieren) und **Export TBX** (als TBX-Datei exportieren). Es werden dieselben Formate unterstützt wie bei den analogen Importfunktionen.

Im oberen Teil des linken Drittels des L/(W-Hauptfensters können Ausgangs- und Zielsprache ausgewählt werden sowie der aktuell zu übersetzende Katalog. Für den aktuellen Katalog wird automatisch angezeigt, wie viele Übersetzungseinheiten bereits übersetzt wurden. Darunter befindet sich das Fenster **UserMessages**, in dem die Übersetzungseinheiten aufgelistet werden. Dabei werden noch nicht übersetzte Einheiten fett dargestellt und bereits übersetzte kursiv. Unterhalb dieses Fensters befindet sich ein Suchfenster, in dem man nach bestimmten Schlüsseln suchen kann.

Oben im rechten Teil des L/(W-Hauptfensters werden die Methoden angezeigt, in denen die aktuelle Übersetzungseinheit vorkommt, und in einem Teilfenster rechts daneben die anderen Übersetzungseinheiten der jeweiligen Methode. Über die Schaltfläche **Open in Class** wird die Klasse geöffnet, in der die Methode implementiert ist (vgl. Abbildung 5.6).

A Localizer	_02
File Import Export	
Source Language Specification: Cjione Method containing selected UserMessage UserMessage in Target Language Specification: de 3 Method containing selected UserMessage UserMessage in Target Language Specification:	this Method:
Katalog 9Sa% (573/538) Katalog 9Sa% (573/538)	Nama Intel I
UserMessages: About/Files: &Files about/WX: About SIVisualWork Accept: Accept Accept: Accept AddChanges. Add &Changes AddChanges. Add &Changes AddChanges. Add &Changes AddClass: &Add Class addDirectory. Add directory AddDirectory. Add the Changes AddConges. Add &to Chan AddToChanges. Add ator AddToChanges. Adj &to Chan AddToChanges. Adj &to Chan Adjg. Topis. FileTools Adj ator atign. FileSelectionChannel currentViewer statusTextRightHolder statusTextLeftHolder view menuBarHolder encodingChannel ' classInstanceVariableNames: ' imports: ' category: Tools-File Browser' Allower Tools.File Browser' Allower Tools.File Browser ' Allower Tools.File Browser '	errHolder refListHolder PaseFile IstHolder err catalogs) catalogy CSEPNe Browser
AllitemsDot: All Berns 67.0% (Dateien) ((Files) translated	in Vis.jolsiidioNET2'>3 and 0 other catalogs)
AmbiguousReferencesExisl: SAmbiguous Referenc AstriFile: &As in File Turicentiane: Allecations	
Albenchmarks: Benchmark runner ATBenchmarks: Benchmarks d	d Accept Translation

Abb. 5.6: Die Klasse, in der die Methode implementiert ist, wird geöffnet

Darunter befindet sich ein Fenster, in dem der Quellcode der die Übersetzungseinheit enthaltenden Methode angezeigt wird. Die Methoden, die mit dem Tool *Canvas* von *VisualWorks* erzeugt wurden, lassen sich in diesem Fenster auch in der WYSIWYG-Darstellung anzeigen (vgl. Abbildung 5.7).



Abb. 5.7: WYSIWYG-Darstellung einer Methode

Im darunter liegenden Teilfenster **Default String** wird die aktuelle ausgangssprachliche Übersetzungseinheit angezeigt, und darunter wiederum befindet sich das Teilfenster **Translations,** in dem die aktuelle Übersetzung, sowie alle zu einem früheren Zeitpunkt vom Übersetzer bestätigten Übersetzungsvarianten, aufgelistet werden. Als zusätzliche Information wird in diesem Teilfenster außerdem angezeigt, wann (Datum und Uhrzeit) die Übersetzung bzw. die Übersetzungsvarianten bestätigt wurde bzw. wurden.

In einem anderen Fenster darunter werden die Übersetzungen angezeigt, die in den importierten Glossaren oder dem Translation Memory gefunden wurden. Dabei kann sowohl die Anzahl der angezeigten Matches reguliert werden (Optionen: Show all, 1, 3, 10, 25, 50, 100) als auch der Matchwert der angezeigten Matches (Spanne: 30% bis 100%).

Am unteren Bildschirmrand befindet sich das Eingabefenster für eine neue Übersetzung und die Schaltfläche Accept Translation, mit der die Übersetzung bestätigt wird und in das systemeigene Translation Memory gespeichert wird.

## 5.4 Evaluierung

## 5.4.1 Allgemeines

L10N ist ein kompaktes Lokalisierungstool, welches in dem speziellen Bereich der Lokalisierung von Smalltalk-Anwendungen einsetzbar ist. Dieser Bereich wird derzeit von keinem anderen Tool abgedeckt. Es besitzt die wichtigsten Funktionen eines Lokalisierungstools, wie die Extraktion des zu übersetzenden Textes einer Anwendung aus dem Programmcode, sowohl zum Zweck der einfacheren Bearbeitung als auch zum Schutz vor unbeabsichtigter Manipulation des Programmcodes, die WYSIWYG-Darstellung des zu übersetzenden Textes innerhalb des Lokalisierungstools (mit Einschränkungen), und damit die Möglichkeit, den Kontext zu einer Übersetzungseinheit zu erfassen, sowie die Möglichkeit, einmal angefertigte Übersetzungen für eine spätere Wiederverwendung nutzbar zu machen. *L10N* läuft auf zahlreichen Plattformen, darunter *Windows, Linux, Unix* und *PowerMac*. Es unterstützt die Translation-Memory- und Terminologie-Austauschformate TMX und TBX, wodurch ein unkomplizierter Datenaustausch mit anderen Translation-Memory-Systemen und Lokalisierungstools gewährleistet wird.

#### 5.4.2 Stärken des Tools

Zu den Stärken des Tools gehört seine Benutzerfreundlichkeit, welche durch die intuitive Benutzeroberfläche erreicht wird. Nach der üblichen Eingewöhnungszeit, die man benötigt, wenn man mit einem unbekannten Tool arbeitet, ist die Bedienung des Tools sehr einfach, da die vorhandenen Funktionen und Optionen auf das Wesentliche beschränkt sind. Die Übereinstimmung der vom System ausgeführten Schritte mit der Erwartung des Benutzers ist aufgrund der gewählten Bezeichnungen der Menüpunkte, Schaltflächen und Labels hoch. Dementsprechend einfach ist der Gebrauch der Anwendung zu erlernen.

Der Status der einzelnen Übersetzungseinheiten und auch der Kataloge ist schnell erkennbar. Die unbearbeiteten Übersetzungseinheiten werden im Teilfenster UserMessages durch fett gedruckte Suchschlüssel dargestellt, die bereits bearbeiteten Übersetzungseinheiten durch kursiv geschriebene Suchschlüssel. Außerdem besteht über das Dialogfeld Settings, erreichbar über den Menüpunkt Options im Menü File, die Möglichkeit, bearbeitete Übersetzungseinheiten auszublenden.

Eine Besonderheit gegenüber anderen Lokalisierungstools innerhalb der WYSIWYG-Ansicht von *L10N* ist die Möglichkeit, zu jedem beliebigen Zeitpunkt des gesamten Übersetzungsprozesses über den Menüpunkt **Original Language** im Menü **File** zwischen Ausgangssprache und Zielsprache zu wählen. Diese Funktion ist sehr hilfreich, um die sprachliche und stilistische Konsistenz der Übersetzung in Anlehnung an den Ausgangstext zu wahren.

Ein Beispiel dafür aus der Lokalisierung des Programms *VisualWorks* ist das Menü **Smalltalk** im Fenster **Workspace**, erreichbar über das Menü **Tools** (vgl. Abbildungen 5.8 und 5.9).

# Dort	Ctrl+D
\$\ Printit	Ctrl+P
Inspect it	Ctrl+Q
\$% Debug it	Ctrl+B
<u>File it In</u>	
Narnespaces	s

Abb. 5.8: Das Menü Smalltalk im Fenster Workspace

4 Ausführen	Ctrl+D
🖗 Printit	Ctrl+P
风 Inspect it	Ctrl+Q
🕬 Debug it	Ctrl+B
<u>F</u> ile it In	
Namespaces	

Abb. 5.9: Das Menü **Smalltalk** in der WYSIWYG-Ansicht von *L10N*, Option Übersetzung, anzeigen

In diesem Fall wurde die Imperativ-Formulierung *Do it* mit dem Infinitiv *Ausführen* übersetzt. Da die Auflistung der Übersetzungseinheiten im Teilfenster **UserMessages** alphabetisch erfolgt, und nicht nach der Reihenfolge ihres Vorkommens im Text, werden die Menüs und Dialogfelder nicht in einem kontinuierlichen Fluss übersetzt. Durch die Möglichkeit, sich auch die bereits überarbeiteten Übersetzungseinheiten desselben Menüs oder Dialogfeldes in der Ausgangssprache anzusehen, wird deutlich, welche Übersetzungsstrategie bei den bisherigen Übersetzungen verfolgt wurde, und diese kann dann auch auf die anderen Übersetzungseinheiten des Menüs oder Dialogfeldes angewendet werden.

Ein weiterer Vorteil des Systems ist bei den vielfältigen Import- und Export-Möglichkeiten zu finden. Dadurch können bestimmte Funktionen, die in *L10N* nicht zur Verfügung stehen, mithilfe anderer Programme ausgeführt werden. Ein Beispiel dafür ist die Pflege des Übersetzungsspeichers. Da beim Bestätigen einer Übersetzung diese automatisch in das Translation Memory gespeichert wird, enthält der Übersetzungsspeicher dann meist auch Übersetzungseinheiten, die zur weiteren Verwendung in anderen Projekten nicht geeignet sind. Durch den TMX- oder TBX-Export können die Daten mit anderen Translation-Memory- oder Terminologieverwaltungssystemen ausgetauscht werden und dort bearbeitet werden (vgl. Abbildung 5.10).

🖺 110n.mdb - SDL Maintain	
TM Edit Format View Iools Help	
🗃 💕 % М5 а 🎮 👗 & е Х 🙏 🗉 🖏 gg в / ш 😵 🧌	2
Filier <no active="" litter=""> d v y l + + w12 • «M</no>	#
i ««Files	1&Dateien
<sup>2</sup> About&VisualWorks	<sup>2</sup>  Über&VisualWorks -
<sup>3</sup> Accept	<sup>3</sup> lÜbernehmen
bhA& P	4 & Hinzufügen
5 Add & Changes	5 RAnderungen hinzufügen
& Add Class	<sup>6</sup> Klasse binzufügen
<sup>1</sup> Add director (	
Add directory	801 line ufiling and
•]&Add	& Hinzufugen
»(Addfile	Datei hinzufugen
"iAdd&File-In	1º 8 <file-1n-dateihinzufügen< td=""></file-1n-dateihinzufügen<>
"(Add &Original Versions	11 & Originalversionen hinzufügen
<sup>12</sup> &AddTo	<sup>12</sup> {»Hinzufügen zu
«AdHoc&SQL	«Ad-hoc-&SQL
<sup>114</sup> Alian Bottom	<sup>14</sup> Unten ausrichten
15 Align Honzontal Center	15 Honzontal mittig ausrichten
16 Align Left	
Alight Leit	
Importing File: DAdiplomarbeitWQrAexportVrranslationsQ5Q7Q5 tmx	4 2
Field J V/JV?	
Created 15.07.2005 22:29:23	
Created By Administrator	
Last Modified 15.07.2005 22:29:23	
Modified Ey Administrator	
Context TEXT	
L	
Hide System fields	
or Helpj press FI	English (United" 5tatesj - > German (Germany)    NUN

Abb. 5.10: Importierte TMX-Datei vonL7(Wnach SDLX

1

## 5.4.3 Verbesserungspotenzial

Aus übersetzerischer Sicht ist die Untergliederung von Übersetzungseinheiten nach Benutzernachrichten als problematisch anzusehen, da innerhalb von *LION* inhaltliche Unterscheidungen bei Homonymen<sup>1</sup> nicht durch unterschiedliche Übersetzungen wiedergegeben werden können. Da bei *LION* pro Suchschlüssel nur eine einzige Übersetzung festgelegt werden kann, muss für ein Homonym eine Übersetzung gefunden werden, die alle Bedeutungsunterschiede der unterschiedlichen Begriffe abdeckt.

Ein Beispiel dafür stellt die Benutzernachricht **Back** dar. Diese Nachricht ist einerseits in der Symbolleiste eines so genannten **Inspektors** enthalten und bedeutet so viel wie *zurück* oder *rückgängig machen* (vgl. Abbildung 5.11).



Abb. 5.11: Elemente der Symbolleiste eines Inspektors (WYSIWYG-Ansicht LION)

Andererseits erscheint die englische Benutzernachricht **Back** aber auch als ein Menüpunkt im Window-Menü eines beliebigen Fensters von *VisualWorks* und bedeutet in diesem Fall so viel wie *nach hinten* (vgl. Abbildung 5.12).

	Relabel äs
	Refresh
	Move
	Resize
	Front
	Back
1	Collapse
-	Close

Abb. 5.12: Elemente eines Window-Menüs

<sup>&</sup>quot;Von Homonymie spricht man, wenn Benennungen sich zwar in ihrer äußeren Form gleichen, die Begriffe, denen sie zugeordnet sind, jedoch [...] keinerlei Ähnlichkeit aufweisen." (Arntz/Picht 1995:135)

Auch die alphabetische Sortierung der Übersetzungseinheiten im Teilfenster UserMessages und die damit verbundene Reihenfolge der Bearbeitung erweist sich als unpraktisch. Es wäre vorteilhafter, wenn es die Möglichkeit geben würde, die Übersetzungseinheiten nach der Reihenfolge ihres Vorkommens im Text zu sortieren und zu bearbeiten. Dann könnten alle Elemente eines Menüs oder Dialogfeldes nacheinander bearbeitet werden, und der Übersetzer müsste sich nicht nach jeder Übersetzungseinheit mit einem anderen Kontext vertraut machen.

Da mithilfe des Suchfensters unterhalb des Teilfensters UserMessages lediglich nach bestimmten Suchschlüsseln gesucht werden kann, nicht aber nach deren Benennungen, lassen sich über diese Funktion nicht alle Übersetzungseinheiten ermitteln.

Ein Beispiel dafür ist der Menüpunkt Class Methods, welcher dem Suchschlüssel ClassDefinitions zugeordnet ist. Gibt der Übersetzer das Suchwort *Methods* ein, wird diese Benutzernachricht nicht gefunden, obwohl die Fuzzy-Match-Suche vom System grundsätzlich unterstützt wird. Da nicht davon auszugehen ist, das der Übersetzer alle Suchschlüssel und die dazugehörigen Benennungen kennt, wäre eine Suchmöglichkeit nach den Bezeichnungen sinnvoller.

Den folgenden Verbesserungsmöglichkeiten ist eine geringere Bedeutung beizumessen, als den bereits genannten.

Das Teilfenster **Default-String**, welches dazu dient, die aktuelle Übersetzungseinheit vollständig darstellen zu können, sollte in der Größe anpassbar sein, damit auch lange Übersetzungseinheiten vollständig dargestellt werden können.

Eine Überprüfung der Zugriffstasten auf ihre Einmaligkeit ist nur durch die WYSIWYG-Darstellung möglich, d. h., bei Elementen, die sich nicht in der WYSIWYG-Ansicht darstellen lassen, kann eine doppelte Vergabe von Zugriffstasten nicht erkannt werden. Deshalb wäre eine Funktion zur automatischen Überprüfung der Zugriffstasten wünschenswert.

Die vom System erstellte Statistik enthält lediglich die Anzahl der Einträge im Translation Memory (einschließlich Glossare) und ist deshalb wenig aussagekräftig bezüglich der Effizienz der Verwendung eines Übersetzungsspeichers. Sehr sinnvoll wäre ein Abgleich der im aktuellen Projekt vorkommenden Übersetzungseinheiten mit den im Translation Memory und in den vorhandenen Glossaren vorkommenden Einheiten. Dadurch könnte vorab festgestellt werden, ob es sich lohnt, ein Translation Memory oder ein Glossar in das aktuelle Übersetzungsprojekt einzubinden.

Für das Projekt verwendete Translation Memories und Glossare sollten angezeigt werden können und auch wieder aus dem Projekt entfernt werden können.

Die Möglichkeit, Kommentare zu den einzelnen Übersetzungen anzulegen, zum Beispiel für Übersetzungsvarianten, wäre eine weitere Verbesserung.

## 5.4.4 Zusammenfassung der Evaluierung

L10N ist ein benutzerfreundliches Lokalisierungstool, welches durch die Unterstützung der standardisierten Datenaustausch-Formate TMX und TBX mit anderen Tools für computergestützte Übersetzung kompatibel ist. Es wurde speziell für die Lokalisierung von Smalltalk-Anwendungen entwickelt, was mit anderen derzeit auf dem Markt erhältlichen Systemen nicht möglich ist. Für zukünftige Versionen des Tools sollte eine Lösung für den Umgang mit homonymen Bezeichnungen gesucht werden. Außerdem wäre es von Vorteil, wenn die Sortierung der Übersetzungseinheiten entsprechend der Reihenfolge ihres Vorkommens im Text möglich wäre. Weitere Optimierungsmöglichkeiten bestehen darin, umfangreichere statistische Angaben zum Projekt anzubieten. Prüffunktionen, wie zum Beispiel Prüfung auf doppelt vergebene Zugriffstasten, eine könnten den ebenfalls vereinfachen Lokalisierungsprozess Gewährleistung bzw. zur der Lokalisierungsqualität beitragen.

# 6 Lokalisierung von *VisualWorks 7.3.1* übersetzungswissenschaftliche Aspekte

## 6.1 Allgemeines

Im Rahmen dieser Diplomarbeit wurden die Menü-Elemente der Benutzeroberfläche von *VisualWorks 7.3.1* mit dem Lokalisierungstool *L10N* übersetzt. Die Übersetzungen wurden zusammen mit ihren Ausgangstexten als TMX-Datei exportiert, welche im Anhang der Diplomarbeit auf einer CD-ROM enthalten ist.

Im nachfolgenden Kapitel werden die übersetzungswissenschaftlichen Aspekte der Lokalisierung von *VisualWorks 7.3.1* vom Englischen ins Deutsche betrachtet. Neben allgemeinen Richtlinien zur Übersetzung von EDV-Texten wird auch auf die Unterschiede zwischen deutschen und englischen EDV-Texten eingegangen. Weiterhin werden mögliche Schwierigkeiten bei der Übersetzung sowie verschiedene Übersetzungsstrategien betrachtet.

## 6.2 Die Übersetzung von EDV-Texten

Die Benutzeroberfläche einer Software lässt sich der Textsorte EDV-Texte zuordnen.

Um den Anforderungen der Benutzer gerecht zu werden und das Kriterium der Benutzerfreundlichkeit zu erfüllen, sollten solche Texte **verständlich** formuliert sein und die Bedienungselemente sollten **einheitlich** beschriftet sein. Der Benutzer sollte nicht überfordert, aber auch nicht unterfordert werden. Wörtliche Übersetzungen von Softwareoptionen sollten vermieden werden. Der Übersetzer sollte immer versuchen, die Bedeutung und die Funktion der Option herauszufinden und dementsprechend eine Übersetzung suchen.

Damit die Terminologie in der Übersetzung **konsistent** ist, sollten keine Synonyme verwendet werden und hersteiler- bzw. produktspezifische Glossare verwendet werden sowie eine projektspezifische Terminologie festgelegt werden.

Auch der sprachliche Stil sollte konsistent sein. Die Konsistenz von Terminologie und sprachlichem Stil sollte nicht nur innerhalb eines einzigen Softwareproduktes gewährleistet werden (zum Beispiel zwischen Menü und Dialogfeld), sondern auch zwischen unterschiedlichen Softwareprodukten beachtet werden sowie im Hinblick auf das zielsprachliche Betriebssystem.

Im Vordergrund steht die Vermittlung von Informationen, daher sollte die Übersetzung **kurz** und **präzise** sein. Sofern es nicht befremdlich wirkt, ist deshalb der Aktiv dem Passiv vorzuziehen. Außerdem sollte der Leser direkt angesprochen werden. Um die Verständlichkeit zu erhöhen und die Bedienung einfacher zu machen, sollte die **logische Reihenfolge** eingehalten werden, zum Beispiel, indem die Übersetzung *aus dem Menü Datei den Befehl* Öffnen der Übersetzung *den Befehl Öffnen aus dem Menü Datei* vorgezogen wird.<sup>1</sup>

Der Benutzer sollte zwar **direkt angesprochen** werden, dies sollte aber möglichst durch die Verwendung des Imperativs erfolgen. Die Verwendung der ersten und zweiten Person sollte vermieden werden, da dies in deutschsprachigen EDV-Texten unüblich ist.

## 6.3 Stil deutscher und englischer EDV-Texte

Der Ausgangstext einer Software ist häufig in amerikanischem Englisch verfasst und auch bei dem Softwaresystem *VisualWorks* ist das der Fall. Solche englischen Texte sind meist in salopper Sprache formuliert, interaktional und adressatenorientiert. Der Adressat wird identifiziert und direkt angesprochen. Vor allem bei Warnungen und Verboten werden Modalverben bzw. Imperativformen eingesetzt. Die **sprachliche Stilebene** deutscher EDV-Texte ist im Allgemeinen jedoch sachlicher und neutraler. Es werden meist unpersönliche Infinitvformen bzw. Lexeme wie *verboten* oder *Warnung* verwendet. Umgangssprache sollte

<sup>&</sup>lt;sup>1</sup> Bezugsquelle: Schmitz/Wähle (2000:101ff)

in der deutschsprachigen Übersetzung vermieden werden.

Nach Stolze (1999:99) stellte Göpferich eine vergleichende Untersuchung zwischen Handlungsanweisungen im Englischen und im Deutschen an. Dabei kam sie zu dem Ergebnis, dass ca. 80% der englischen Anweisungen im direktiven Imperativ verfasst sind, wie zum Beispiel *Push the red button*. Im Deutschen wird demnach zu annähernd 60% der imperativische Infinitiv oder die Infinitivellipse wie zum Beispiel bei *Schlüssel abziehen* verwendet. Der direktive Imperativ wie zum Beispiel *Deponieren Sie keine schweren Gegenstände aufder Hutablage* wird im Deutschen in nur etwa 11% der Fälle verwendet.

Auch Gratulationen, die im Englischen vorkommen, sollten nicht in die deutsche Übersetzung einfließen. *Congratulations, you have successfully installed the application* könnte demnach zum Beispiel einfach mit *Sie haben die Anwendung erfolgreich installiert* übersetzt werden.

Englische Texte sind im Allgemeinen weniger **differenziert** als deutsche, besonders was die Wortwahl betrifft. Das zeigt sich unter anderem darin, dass im Englischen eine geringere Anzahl an unterschiedlichen Verben verwendet wird und mehr einfache Verben als im Deutschen gebraucht werden, wie zum Beispiel *be, do, go, have, make* oder *use*.

Beispiele dafür finden sich in *VisualWorks* bei den Befehlen *Do it* oder bei *Get info*, welche mit *Ausführen* bzw. mit *Informationen anzeigen* übersetzt wurden.

Im Deutschen werden englische Aktivkonstruktionen häufig durch Passivkonstruktionen wiedergegeben. Um jedoch den Benutzer einer Software direkt anzusprechen, sollte in der deutschsprachigen Übersetzung einer Benutzeroberfläche ebenfalls der Aktiv verwendet werden. Dies gilt jedoch nicht, wenn unbelebte Objekte so stark personifiziert werden, dass es für den Benutzer befremdend wirkt.

Dass der Benutzer um etwas gebeten wird, ist ein typisches Merkmal von englischsprachigen EDV-Texten. Für deutschsprachige EDV-Texte ist dies aber eher untypisch. Es gibt allerdings auch Situationen, in denen auch im Deutschen eine **Bitte** formuliert werden sollte, nämlich wenn der Benutzer durch das Ausführen einer Aktion eine längere Zeit warten muss.

Eine **Besitzanzeige** durch Possesivpronomen ist in deutschsprachigen EDV-Texten oft überflüssig und wirkt störend, wenn sie in eine Übersetzung übernommen wird. Unter bestimmten Bedingungen ist sie jedoch erforderlich, zum Beispiel bei der Übersetzung von *Please typeyour name: Bitte geben Sie Ihren Namen ein.* 

Auch die **Zeichensetzung** muss den zielsprachlichen Konventionen angepasst werden. Dies gilt besonders für Ausrufezeichen, welche im Englischen im Allgemeinen weniger häufig verwendet werden als im Deutschen. Ausrufezeichen werden im Englischen meist nur in Zusammenhang mit einem tatsächlichen Ausruf verwendet oder wenn Verwunderung zum Ausdruck gebracht werden soll<sup>2</sup>. Im Deutschen wird das Ausrufezeichen darüber hinaus auch bei einer nachdrücklichen Feststellung, einer Aufforderung und einer direkten Anrede gebraucht<sup>3</sup>.

Ebenso sollte die **Großschreibung** angepasst werden. Im Englischen werden Elemente der Benutzeroberfläche großgeschrieben. Da im Deutschen auch alle Substantive großgeschrieben werden, müssen Elemente der Benutzeroberfläche anderweitig hervorgehoben werden. Das kann zum Beispiel durch Fettdruck oder Anführungszeichen realisiert werden.

# 6.4 Schwierigkeiten bei der Übersetzung von EDV-Texten

Ein häufiges Problem bei der Übersetzung von EDV-Texten stellt die **Mehrdeutigkeit** von einzelnen Wörtern und Phrasen dar.

Diese Mehrdeutigkeit kann einerseits durch Homonymie entstehen, also dadurch, dass mehrere Begriffe die gleiche Benennung haben können. Zur Unterscheidung zwischen Begriff und Benennung geht Stolze (1997:43) auf die Auffassung von Ferdinand de Saussure ein.

<sup>&</sup>lt;sup>2</sup> Bezugsquelle: http://en.wikipedia.org/wiki/Exclamationmark, Zugriff: 08.08.2005

<sup>&</sup>lt;sup>3</sup> Bezugsquelle: http://de.wikipedia.org/wiki/Ausrufezeichen, Zugriff: 08.08.2005

Demnach

"besteht jedes Sprachzeichen aus den zwei Aspekten Ausdruck/Inhalt, also aus einem materiellen (lautlich oder graphisch) realisierten Zeichenkörper oder Signifikanten (*signiflant*) und dem Zeicheninhalt, dem begrifflichen Konzept als Signifikat (*signifié*)."

Die grafische Darstellung (vgl. Abbildungen 6.1 und 6.2) wird bilaterales bzw. zweiseitiges Zeichenmodell genannt<sup>4</sup>.



Abb. 6.1: Bilaterales Zeichenmodell, allgemein



Abb. 6.2: Bilaterales Zeichenmodell, Beispiel "Haus"

Andererseits können unterschiedliche grammatikalische Erscheinungen mit ein und derselben äußeren Form eines Wortes ausgedrückt werden.

Ein Beispiel für diese beiden Phänomene aus der Lokalisierung des Programms *VisualWorks* ist das Wort *matching* aus dem Menüpunkt **Class Matching** im Menü **Tools**, Untermenü **Change List**, Untermenü **Find**. Einerseits kann *matching* die Bedeutung *Abgleich, Vergleich, Gleichheitsprüfung* haben, andererseits kann es auch die Bedeutung *Anpassung* haben. Als mögliche Übersetzungen kommen also ein *Klassenabgleich* oder eine *Klassenanpassung* in Frage. Aus dem Kontext ergibt sich jedoch, dass keine dieser beiden Lösungen zutreffend ist. Denn in diesem Fall liegt eine Verkettung von mehreren Bezeichnungen innerhalb eines Menüs vor. *Matching* ist in diesem Kontext kein Nomen, sondern ein Partizip Präsens. Der vollständige Befehl lautet: **Find Class Matching Instance and Class/Selected Class Only** und hinter dem Wort *matching* verbirgt sich, dass eine Klasse gesucht werden soll, auf die eine bestimmte Bedingung zutrifft. Um eine Umschreibung durch

<sup>&</sup>lt;sup>4</sup> Bezugsquelle: http://www.imi4rier.de/uni/ft2/ldv/ldvwiki/index.php/FerdinanddeSaussure, Zugriff: 08.08.2005
einen Relativsatz zu umgehen, wurde in der deutschen Übersetzung die Wiedergabe von *matching* durch drei Auslassungspunkte gewählt, damit der Benutzer erkennt, dass zum Ausführen des Befehls weitere Optionen ausgewählt werden müssen.

Ein weiteres Übersetzungsproblem stellen **Anglizismen** dar. Als Anglizismen werden Begriffsbezeichnungen oder sprachliche Eigentümlichkeiten bezeichnet, die vom Englischen in eine andere Sprache übernommen werden. Oftmals wäre bei Anglizismen eine Übersetzung ins Deutsche möglich, aber in Fachsprachen, und besonders in der Fachsprache der Informatik, werden häufig die englischen Bezeichnungen zu Gunsten der Eindeutigkeit bevorzugt.

"[...] lieber eine eindeutige englische Bezeichnung, bei der man weiß, was gemeint ist, als eine oder sogar verschiedene krampfhafte deutsche Übersetzungen, die keiner versteht." (Schmitz/Wähle 2000:114)

Bei der Lokalisierung des Programms *VisualWorks* trifft dies auf den Befehl **Debug** zu. Die wörtliche deutsche Übersetzung ist *entwarnen*, eine mögliche Übersetzung könnte auch durch eine Umschreibung wie *Fehler suchen und beseitigen* erfolgen. In der Fachsprache der Informatik wird für diesen Begriffjedoch üblicherweise die englische Bezeichnung verwendet und eine Übersetzung ins Deutsche könnte dazu führen, dass der Benutzer mit der Übersetzung nichts anzufangen weiß.

Besondere Beachtung sollte auch der Übersetzung von **Komposita** gelten, da die Bezüge der einzelnen Bestandteile zueinander im Englischen nicht durch Präpositionen oder ähnliches ausgedrückt werden. Enthält eine Wortgruppe mehrere Substantive und ein Adjektiv, besteht die Gefahr, dass das Adjektiv dem falschen Substantiv zugeordnet wird, da die Zuordnung eines Adjektivs zu einem Substantiv formal nicht zu erkennen ist. Eine richtige Lösung für ein solches Problem kann nur durch Sachkenntnis erzielt werden.

Der Menüpunkt Exported Class References in VisualWorks könnte demnach rein formal mit Exportierte Klassenverweise übersetzt werden, oder auch mit Verweise auf exportierte Klassen. Da class references jedoch ein feststehender Begriff bei der Programmierung mit objektorientierten Sprachen, und somit auch in VisualWorks, ist, kann daraus abgeleitet werden, dass Exportierte Klassenverweise in diesem Kontext eine richtige Übersetzungslösung ist.

Die Änderung der Gestalt eines Wortes zum Ausdruck seiner grammatischen Funktion wird als **Flexion** bezeichnet. Das heutige Englisch weist nur noch wenige Flexionen auf. Daraus kann sich ein Übersetzungsproblem ergeben, da die Wortkategorie formal manchmal nicht zu erkennen ist. Die Lösung kann in diesem Fall nur über den Kontext ermittelt werden, welcher allerdings bei Benutzeroberflächen aus Platzgründen meist nur in sehr geringem Umfang vorhanden ist.

Verben können im Deutschen im Gegensatz zum Englischen zum Beispiel nach Person, Numerus, Aspekt, Zeit, Genus verbi oder Modus konjugiert werden.

Ein Beispiel aus der Lokalisierung des Programms *VisualWorks* ist der Menüpunkt **Inspecting Objects** im Menü **Tools**, Untermenü **Help**. Handelt es sich hierbei um ein Kompositum aus einem Gerundium und einem Nomen, dann wäre eine mögliche Übersetzung *Inspizierungsobjekte*. Es kann aber auch ein eigenständiges Nomen mit einem eigenständigen Gerundium sein, dann wäre die Übersetzung *Objekte inspizieren* denkbar. Eine dritte Möglichkeit ist, dass ein Partizip Präsens gemeint ist, dann könnte *inspizierende Objekte* eine Lösungsmöglichkeit darstellen.

Deutsche Nomina werden nach Kasus, Numerus oder Genus dekliniert, nicht so im Englischen. Deshalb kann der Menüpunkt **Format Selection** ein Kompositum aus zwei Nomen sein, dann könnte er mit *Formatauswahl* übersetzt werden. Oder es könnte ein Infinitv mit einem Nomen sein, dann wäre *Auswahlformatieren* eine mögliche Übersetzung.

Wird der Menüpunkt **Select one** isoliert betrachtet, ist das Bezugswort für *one* nicht feststellbar und es ergeben sich die Übersetzungsmöglichkeiten *Einen auswählen, Eine auswählen* oder *Eins auswählen*. Die richtige Lösung kann wieder nur durch den Kontext ermittelt werden. Sollte der Kontext bei flexionsbedingten Übersetzungsproblemen nicht ermittelbar sein, sollte möglichst eine nicht zu flektierende Übersetzung gewählt werden, zum Beispiel *ohne, kein Eintrag* oder *keine Angabe* für *none*.

Eine geringe **Motiviertheit** des ausgangssprachlichen Wortes kann ein weiteres Übersetzungsproblem darstellen.

"Die Motiviertheit eines morpholog. komplexen Wortes kennzeichnet das Ausmaß, in dem es sich als Summe der Bedeutungen seiner Teile und der Weise ihrer Zusammenfügung verstehen läßt." (Metzler Lexikon Sprache 1993:405) Ein Beispiel aus der Lokalisierung des Programms *VisualWorks* dafür ist der Befehl **Step.** Mögliche Übersetzungen für *step* sind *Schritt* bzw. *gehen, Stufe* bzw. *abstufen* oder *Sprung* bzw. *springen*. Die Funktion des Befehls ist nicht anhand seiner Bezeichnung zu erkennen, denn seine Bedeutung wird wie folgt beschrieben:

"Führt ein vollständiges Nachrichtensenden in der gewählten Methode durch, d.h. die gesamte Methode, die der hervorgehobenen Nachricht entspricht, wird ausgewertet." (Hopkins/Horan 1997:271)

"Der Unterschied zwischen **step** und **send** besteht darin, daß mit **Step** die im Programmfenster markierte Nachricht gesendet und anschließend die gesamte dadurch aktivierte Methode ausgeführt wird." (Bücker et al. 1995:133)

Abkürzungen können im Ausgangstext in Form von fachspezifisch genormten Standardkürzeln vorliegen, die zum Teil auch sprachübergreifend genormt sind, oder aber auch als so genannte Ad-hoc-Kürzel, die aufgrund ihrer Ungebräuchlichkeit das Verständnis erschweren können. Ein solches Beispiel findet man in *VisualWorks* bei dem Menüpunkt Sel'n Frgnd (Selecting Foreground) im Menü Colors, Untermenü Policy Colors.

Die Elemente einer Benutzeroberfläche werden im Englischen großgeschrieben. Die Benutzeroberfläche von *VisualWorks* ist jedoch inkonsistent bei der Verwendung von Großbuchstaben. Es kommt vor, dass alle Wörter eines Menüpunkts großgeschrieben werden, unabhängig davon, ob es sich um Elemente der Benutzeroberfläche oder Eigennamen handelt. So gibt es in *VisualWorks* zum Beispiel den Menüpunkt Copy Name ebenso wie den Menüpunkt Copy name. Dies erschwert die Übersetzung dahingehend, dass es für den Übersetzer unter Umständen schwierig sein kann, Eigennamen und Elemente der Benutzeroberfläche zu identifizieren. Zum Beispiel wäre für den Menüpunkt As in File neben der Übersetzung *Wie in Datei* formal auch die Übersetzung *Wie im Menü Datei* möglich, was inhaltlich ein großer Unterschied ist.

Eine deutsche Übersetzung ist durchschnittlich 15% **länger** als das englische Original (siehe Kapitel 3.2). Alle Elemente der lokalisierten Benutzeroberfläche müssen jedoch bei jeder Bildschirmauflösung auf den Bildschirm passen, deshalb sollte die Übersetzung allgemein so kurz wie möglich sein. Dies kann unter Umständen auch durch Abkürzungen realisiert werden, wobei auch wieder auf Konsistenz geachtet werden muss, auch an Stellen,

wo genügend Platz für das ausgeschriebene Wort wäre.

Eine Zugriffstaste, auch Tastaturkurzbefehl genannt, bietet eine alternative Möglichkeit, um auf Menübefehle oder Dialogfeld-Optionen zuzugreifen. Dies kann auf der Benutzeroberfläche zum Beispiel durch einen unterstrichenen Buchstaben dargestellt werden. Der Befehl wird aktiviert, wenn dieser Buchstabe zusammen mit der Alt-Taste gedrückt wird. Bei der Übersetzung ist zu beachten, dass möglichst keine Buchstaben mit Unterlänge dafür ausgewählt werden, da die Unterstreichung auf dem Bildschirm sonst kaum sichtbar ist. Eine Zugriffstaste kann nur einmal pro Menü bzw. Untermenü oder Dialogfeld-Option vergeben werden. Es sollte auf Konsistenz der Zugriffstasten in Bezug auf das zielsprachliche Betriebssystem geachtet werden, besonders bei Standardbefehlen wie *Öffnen, Speichern* oder *Drucken*.

## 6.5 Übersetzungsstrategien

Übersetzungen können in Anlehnung an die Unterscheidung in wörtliche und freie Übersetzung auch nach verfremdender oder einbürgernder Übersetzung unterschieden werden (Snell-Hornby et al. 1998:151). Bei der Einbürgerung wird der Text an die Normen der Zielsprache angepasst, bei der Verfremdung wird er nicht angepasst. Die Wahl der Übersetzungsmethode hängt vom Texttyp und von der Übersetzungsfunktion ab. Bei einem hohen Grad an Fachlichkeit, was bei der Softwarelokalisierung zutrifft, liegt oft eine sehr spezifische Übersetzungsfunktion vor, deshalb wird dabei meist die sprachlich und kulturell einbürgernde Übersetzung gewählt.

Von der Übersetzungsmethode wird das Übersetzungsverfahren unterschieden. Die Übersetzungsmethode bezieht sich auf den gesamten Text und ist abhängig vom Texttyp und von der Übersetzungsfunktion, das Übersetzungsverfahren hingegen bezieht sich auf kleinere Textabschnitte und ist abhängig von der Übersetzungsmethode und dem Sprachen- bzw. Kulturpaar.

Im Folgenden werden verschiedene Verfahren der Textübersetzung erläutert, die bei der Lokalisierung der Benutzeroberfläche von *L10N* zur Anwendung gekommen sind.<sup>5</sup>

<sup>&</sup>lt;sup>5</sup> Auf der Grundlage von Snell-Hornby et al. 1998:151f

Bei der **lexikalischen Entlehnung** wird eine lexikalische Einheit von der Ausgangssprache in die Zielsprache übernommen und an die zielsprachliche Grammatik angepasst. Typische Beispiele für lexikalische Entlehnungen sind die Übernahme von Realia<sup>6</sup>.

Bei der Lokalisierung der Benutzeroberfläche von *L10N* wurde der Terminus *Override* ins Deutsche übernommen, da er einen für die Smalltalk-Programmierung typischen Befehl umschreibt, für den es keine deutsche Übersetzung gibt. Es handelt sich hierbei um den Befehl, eine Methode zu überschreiben, wobei die neu entstandene Methode zukünftig Anwendung findet, die überschriebene Methode jedoch abgespeichert wird und so bei Bedarf auf diese wieder zugegriffen werden kann. Die Möglichkeit, den Terminus *Override* durch eine Umschreibung zu übersetzen, wurde bewusst nicht verfolgt, da die Übersetzungen von Benutzeroberflächen von Software starken Längenbeschränkungen unterliegen.

Die **lexikalische Ersetzung**, auch Substitution genannt, bedeutet die Ersetzung eines lexikalischen Elementes der Ausgangsprache durch ein zielsprachliches Element. Dieses Übersetzungsverfahren wird auch als Trivialfall bezeichnet.

Beispiele aus der vorliegenden Lokalisierung sind die Befehle **Copy** und **Add**, die mit *Kopieren* und *Hinzufügen* übersetzt wurden.

Bei der **Wort-für-Wort-Übersetzung** werden in der Übersetzung die Wortzahl, die Wortart und die Wortstellung des Ausgangstextes beibehalten. Dieses Verfahren ist ebenfalls ein relativ einfaches Übersetzungsverfahren.

Beispiele aus der Lokalisierung des Programms *VisualWorks* sind hierfür die Menüpunkte About VisualWorks, der mit *Über VisualWorks* übersetzt wurde, und As in File, der mit *Wie in Datei* übersetzt wurde.

Bei der **Permutation** handelt es sich um die Umstellung von Konstituenten, das kann zum Beispiel die Anwendung der deutschen Satzklammer zur Wiedergabe eines Satzes mit der typischen englischen Wortstellung Subjekt-Prädikat-Objekt sein.

Ein bei der Lokalisierung von *VisualWorks* verwendetes Beispiel dafür ist der Befehl Add Changes, der mit *Änderungen hinzufügen* übersetzt wurde. Dieses Übersetzungsverfahren wurde hauptsächlich bei der Übersetzung mit Infinitven unter

<sup>&</sup>lt;sup>6</sup> "Element des Alltags, der Geschichte, der Kultur, der Politik u.drgl. eines bestimmten Volkes, Landes, Ortes, die keine Entsprechung bei anderen Völkern, in anderen Ländern, an anderen Orten hat, [...]" (Snell-Hornby et al. 1998:288)

Einhaltung der M/craro/z'-Konventionen angewandt.

Bei der **Expansion bzw.** der **Reduktion** wird die Wortzahl der Übersetzung im Vergleich zum Ausgangstext erhöht bzw. verringert.

Dieses Verfahren wurde zum Beispiel bei der Übersetzung des Befehls **Print it** angewandt, was unter Auslassung des Pronomens allein durch das Verb *drucken* übersetzt wurde, da eine Übernahme des Pronomens *es*, welches dem englischen *it* entspricht, beim deutschsprachigen Rezipienten mit größter Wahrscheinlichkeit Verwunderung hervorgerufen hätte, da die Formulierung *Drucke es* nicht den deutschsprachigen Konventionen bei der Bezeichnung von Benutzeroberflächen von Software entspricht.

Bei dem Übersetzungsverfahren der **Transposition** wird eine Änderung der Wortart vorgenommen.

Dies fand zum Beispiel bei dem Befehl **Align bottom** statt, was mit *Unten ausrichten* übersetzt wurde. Hier wurde das englische Substantiv *bottom* durch das deutsche Adverb *unten* ersetzt, um wiederum das Kriterium der Kürze zu erfüllen. Eine längere Übersetzungsvariante, wie *Am unteren Rand ausrichten* oder *An (der) Unterkante ausrichten*, konnte dadurch vermieden werden.

# 7 Fazit und Perspektiven

Der Lokalisierungsprozess von Softwareprodukten kann durch eine Vielzahl von Systemen und Technologien unterstützt werden. Das Lokalisierungstool *L10N* füllt dabei eine Lücke im Bereich der computergestützten Softwarelokalisierung aus, da mit anderen derzeit erhältlichen Lokalisierungstools keine Smalltalk-Anwendungen lokalisiert werden können. Es besitzt die wichtigsten Funktionen eines typischen Lokalisierungstools, wie sie derzeit auf dem Markt erhältlich sind, und verfügt über eine intuitive Benutzeroberfläche. Bemühungen zur Weiterentwicklung bzw. Verbesserung des Tools sollten vor allem dahingehen, dass für gleiche Benennungen in der Ausgangssprache unterschiedliche Benennungen in der Zielsprache möglich sind, und dass die Sortierung der Übersetzungseinheiten nach der Reihenfolge ihres Vorkommens im Text möglich ist.

Eine vollständig lokalisierte Version von *VisualWorks 7.3.1* könnte besonders als Schulungsversion für Programmieranfänger mit geringen Englischkenntnissen genutzt werden. Für versierte Programmierer kann die Arbeit mit der lokalisierten Programmversion eine Umstellung bedeuten, da Entwicklungsumgebungen (im Gegensatz zu Anwendungssoftware) meist in einer englischsprachigen Version vorliegen und Fachleute deshalb den Umgang mit englischsprachigen Bezeichnungen gewohnt sind.

# Literaturverzeichnis

- Arntz, Reiner/Picht, Herbert (1995): Einführung in die Terminologiearbeit. Georg Olms Verlag, Hildesheim [etc.].
- Bücker, Matthias C. et al. (1995): Programmieren in Smalltalk mit VisualWorks®. Springer-Verlag, Berlin [etc.].
- Brauer, Johannes (2003): Grundkurs Smalltalk Objektorientierung von Anfang an. Viehweg Verlag, Wiesbaden.
- Duden (2001): Deutsches Universalwörterbuch. Bibliographisches Institut & F. A. Brockhaus AG, Mannheim.
- Duden (1998): Grammatik. Bibliographisches Institut & F. A. Brockhaus AG, Mannheim.
- Duden (2001): Richtiges und gutes Deutsch. Bibliographisches Institut & F. A. Brockhaus AG, Mannheim.
- Eisenberg, Peter (1999/2001): Grundriß der deutschen Grammatik. Band 1: Das Wort. Verlag J. B. Metzler, Stuttgart [etc.].
- Eisenberg, Peter (1999/2001): Grundriß der deutschen Grammatik. Band 2: Der Satz. Verlag J. B. Metzler, Stuttgart [etc.].
- Esselink, Bert (2000): A Practical Guide to Localization. John Benjamins Publishing Company, Amsterdam [etc.].
- Fissgus, Ursula/Seewald-Heeg, Uta (2005): "Softwarelokalisierung", in: Information Technology 4/2005, 220-225.
- Gercken, Jürgen (1999): Kultur, Sprache und Text als Aspekte von Original und Übersetzung. Peter Lang GmbH, Frankfurt am Main.
- Gerzymisch-Arbogast, Heidrun (1994): Übersetzungswissenschaftliches Propädeutikum. A. Francke Verlag, Tübingen [etc.].
- Honig, Hans G. (1997): Konstruktives Übersetzen. Stauffenburg Verlag Brigitte Narr GmbH, Tübingen.

- Hopkins, Trevor/Horan, Bernard (1997): Objektorientierte Programmierung mit Smalltalk. Carl Hanser Verlag, München [etc.].
- Horn-Helf, Brigitte (1999): Technisches Übersetzen in Theorie und Praxis. A. Francke Verlag, Tübingen [etc.].
- Koller, Werner (1997): Einführung in die Übersetzungswissenschaft. Quelle & Meyer Verlag GmbH & Co., Wiesbaden.
- Metzler Lexikon Sprache (1993). Verlag J. B. Metzler, Stuttgart [etc.].
- Microsoft Press (1999): Computer-Fachlexikon mit Fachwörterbuch. Microsoft Press Deutschland, Unterschleissheim.
- Moreaux, Marie-Anne (2003): Systèmes de codage des caractères. Begleitmaterial zur Vorlesung « Principes de fonctionnement des ordinateurs » des Studienganges « Licence de Traitement Automatique des Langues ». INaLCO, Paris.
- Reineke, Detlef/Schmitz, Klaus-Dirk (Hrsg.) (2005): Einführung in die Softwarelokalisierung. Narr Francke Attempto Verlag GmbH + Co. KG, Tübingen.
- Schmitz, Klaus-Dirk/Wahle, Kirsten (Hrsg.) (2000): Softwarelokalisierung. Stauffenburg Verlag Brigitte Narr GmbH, Tübingen.
- Snell-Hornby, Mary et al. (Hrsg.) (1998): Handbuch Translation. Stauffenburg Verlag Brigitte Narr GmbH, Tübingen.
- Somers, Harold (2003): Computers and Translation. John Benjamins Publishing Company, Amsterdam [etc.].
- Stolze, Radegundis (1997): Übersetzungstheorien. Gunter Narr Verlag, Tübingen.
- Stolze, Radegundis (1999): Die Fachübersetzung. Gunter Narr Verlag, Tübingen.
- Symmonds, Nick (2002): Internationalization and Localization Using Microsoft .NET. Springer Verlag GmbH & Co. KG, Heidelberg.

## **Online im Internet**

- Catalyst-Hompage. http://www.alchemysoftware.ie/index.html [Zugriff: 10.07.2005]
- Cincom-Homepage. http://smalltalk.cincom.com/prodinformation/index.ssp [Zugriff: 12.07.2005]
- Englisch-Deutsches Wörterbuch. http://branchenportal-deutschland.aus-stade.de/ei3.htm [Zugriff: 20.04.2005]
- Fachwörter Englisch-Deutsch. Online-IT-Lexikon, http://www.itwissen.info/index.php?letter=U&id=57 [Zugriff: 04.06.2005]
- Forum Deutsches Recht. http://www.recht.de/index.php3?menue=Rechtsgebiete&linkJiat\_id=562&linkkat=152&kn ame=Wettbewerbsrecht [Zugriff: 06.08.2005]
- Lion'05. Seite zur 2. Kölner Tagung zum Thema Softwarelokalisierung. http://www.iim.fh-koeln.de/lion2005/ [Zugriff: 12.07.2005]
- Lokalisierungsrichtlinien für die Benutzeroberfläche. http://msdn.microsoft.com/library/deu/default.asp?url=/library/DEU/modcore/html/deconlo calizationconsiderationswithuserinterface.asp [Zugriff: 03.05.2005]
- Passolo-Homepage. http://www.passolo.com/ [Zugriff: 02.07.2005]
- Pocket Smalltalk User's Guide. http://www.pocketsmalltalk.com/pst-doc-ug-3.html [Zugriff: 14.04.2005]
- Policy-basiertes Management von verteilten Systemen. http://www.fernuni-hagen.de/DVT/StudiumLehre/Arbeiten/krell.pdf [Zugriff: 22.04.2005]
- SAM. Deutscher Vertriebspartner von Catalyst. http://www.sam-engineering.de/site.php?lang=de&site=start&class=start [Zugriff: 10.07.2005]

[Zugriff: 11.05.2005]

Scriptum C++2/2. http://www.ifs.uni-linz.ac.at/~sre/c++/ScriptO2.html [Zugriff: 16.04.2005] SDL-Homepage. http://www.sdl.com/ [Zugriff: 19.07.2005] Smalltalk. http://www.netzwelt.de/lexikon/Smalltalk\_(Programmiersprache).html [Zugriff: 26.04.2005] Softguide. Marktübersicht für Softwareprodukte, http://www.softguide.de/ [Zugriff: 12.07.2005] Star-Homepage. http://www.star-group.net/deu/home.html [Zugriff: 16.07.2005] TEMPO-Handbuch. http://vhb.fh-regensburg.de/wep/map/material/handbuch.pdf [Zugriff: 14.04.2005] Trados-Internetseite. http://www.translationzone.com/ [Zugriff: 19.07.2005] Verlust der Flexion in der englischen Sprache. http://www.weikopf.de/Sprache/Englisch/Verlust Flexion/verlust flexion.html [Zugriff: 11.05.2005] VisualWorks für Linux: Eine industrietaugliche Smalltalk-Entwicklungsumgebung, http://www.lugg.de/pub/misc/linuxtag\_1 999/html/VW\_Linux.html [Zugriff: 15.04.2005] WikiLingua: Ferdinand de Saussure. http://www.uni-trier.de/uni/fb2/ldv/ldv\_wiki/index.php/Ferdinand\_de\_Saussure [Zugriff: 08.08.2005] Wikipedia (freie Enzyklopädie). http://de.wikipedia.org/wiki/Hauptseite

# Anhang

## The localization tool L10N

## 1 Introduction

Currently available localization tools are mainly designed for the localization of *Windows* programmes. Therefore, the Company *Georg Heeg eK* developed the localization tool *L10N*, so that applications created with *VisualWorks* can easily be localized, too.

L10N can be run on all platforms VisualWorks is running. These are namely Windows (95/98/ME/NT/2000), PowerMac, Intel Linux, AIX, SGI Irix, Compaq UNIX, HP-UX, and Solaris.

Unlike other localization tools and translation memory Systems, *L10N* does not create a separate file for the localization project. Instead, the data accumulated in the translation memory is stored in a *VisualWorks* image file. The data accumulated in the translation memory can be stored in an export file by exporting them.

The entire text of an application to be translated is the totality of all texts seen by the user. In *VisualWorks*, those texts are called user messages. The subdivision into translation units is made accordingly to the user messages. Each user message represents a translation unit.

## 2 The user interface

The localization tool *LION* is implemented in the *VisualWorks 7.3.1* environment (see Figure A.1). It can be opened by selecting the first icon on the right in the toolbar, a blue book, or in the **Window** menu the menu item **1.** [FUI] Localizer.



Fig. A.1: The VisualWorks launcher with the LION icon

The localization tool will be opened in a separate window, which is shown in Figure A.2.

File Import Export   Source Language Specification: c	🔄 Localizer								X
Source Language Specification:    Target Language Specification:	File Import Export								
Target Language Specification: de   Marget Language Specification: de   Katalog: 0.7% (4/548) labei:   UserMessages: Abort: Abort Distribution:   Abort: Abort Source Visual   Accessors Accept Accessors Accept Accessors Accept   Add:made form: Add Watton Add Marget Add	Source Language Specification:	c_none	Method contain	ning selected UserMessage	UserMessages inthis Method:				
Katalog. [0.7% (4/548) labeis JuserKessages:   UserKessages: Journal (4/548)   Abort: Abort aboutVisualWorks About Visual Works Accessors Accessors Isource Visual /   Accessors Accessors "UPainter new openOnClass: self andSelector: WindowSpec"   AddSaudSubton: addButton addButton: addButton AddSubCarvas: AddSubCarvas "###ULFullSpec]   AdvancedUtilities: Advanced Utilities AdvancedUtilities: Advanced Utilities AdvancedUtilities: Advanced Utilities AdvancedUtilities: Advanced Utilities AdvancedUtilities Advanced AllowEarrelia Alphabetical AllowEarrelia Alphabetical Application Application Development: Application Development Matchine Front Astouteneatindent Assign To Named Ford Advancestifyen (Ad	Target Language Specification:	de 2	windowSpec	in UI.SimpleListEditord	Add. Add Change: Change				1
UserMessages:   Abort: Abort   Abort: Abort   aboutVisualWorks: About Visual Works   Accessors: Accessors   Accessors: Accessors   Add: Add   addButton: addButton   addButton: addButton   addButton: addButton   AdvancedUtilities: Advanced Utilities   AdvancedUtilities: A	Katalog.   0.7% (4/548) la	beis	d		Delete: Delete				-
Abort: Abort windowSpec 5   Accessors Accessors Accessors Consors <td>UserMessages:</td> <th></th> <td>  Source! Vis</td> <td>sual  </td> <td></td> <td></td> <td></td> <td></td> <td></td>	UserMessages:		Source! Vis	sual					
Allow Settings: Allow Continance-Line Settings (set Alphabetical altPlusString: Alt+<1s>   Anpabetical altPlusString: Alt+<1s>   AmpHelp: feHslp   AnErrorOccurredInSetting an error has occurred ApplicationDevelopment: Application Development ApplicationDevelopmentDesc Tools and framewi ApplicationServer Application Server ApplicationServer ApplicationServer Polication Server ApplicationServer Server ApplicationServer Application Server ApplicationServer ApplicationServer Application Server ApplicationServer Ap	Abort: Abort aboutVisualWorks About Visual V Accept: Accept Accessors Accessors Acd Add addButton: addButton AddNamedFont: Add New Named AddSubCanvas: AddSubCanvas AdvancedCDLabel: Advanced Utilit AdvancedUtilities: Advanced Utilities AdvancedUtilities: AdvancedUtilities AdvancedUtilities: Adv	/orks Font es ubtle and wizari J-Line Expressi Tie-ins (fiein, 1 and-Line Parce	windowSpec "UlPaint <resour ##(UL #vvi #(# Default-String Translations</resour 	er new openOnClass: seif ce: #canvas> FullBpec} ndow: UI. WindowSpec} #label: #(#[Kernel.UserMessage) # Add Translator - 31. Juli 2005 Translator - 31. Juli 2005 Translator - 31. Juli 2005	andSelector: WindowSpec" 19:55:00: Hinzufügen 19:51:57: Hinzufügen 19:51:53: Hinzufügen 19:43:39: Hinzufügen				5 
AnErrorOccurredInPageDefinition An error has i AnErrorOccurredInSetting: An error has occurred Application Application Development: Application Development ApplicationServer Application Server ApplicationServer	Alphabetical Alphabetical altPlusString: Alt+<1s>		Existing Trans	lations	Mumberto show	: 20 J^	Matching:	70	-3%
AssigiiNamedFant Assign To Named Font AutoInstallWbenDefining: Auto-Install when defini AutoInstallWbenDefining: Auto-Install when defini	AnErrorOccurredInPageDefinition AnErrorOccurredInSetting An erro Application: Application ApplicationDevelopment: Applicat ApplicationDevelopmentDesc To ApplicationServer Application Ser ApplicationServerDesc VisualWo aRepositor/ a Repository	An error has i or has occurred ion Developrnei ols and framewi ver rks Web Applic	9B.0%   Hinz     98.0%   Hinz     96.0%   Addii     73.0%   AHin     71.0%   Hinz     71.0%   Hinz	ufügen ulegen eren zufügen ufügen» ufügen:	(Add translated in labeis and 9 (Add translated in labeis and 0 (Add translated in E>.:12M5 ar (SAdd translated in menus and (Add? translated in WindowsSt (Add: translated in WindowsXP	D other catalogs) other catalogs) ad 2 other catalogs 0 other catalogs aruerIOH and 1 c and 0 other cata	) s) other catalog) alogs)		J
	AssigiiNamedFant Assign To Narr AutoInstallWbenDefining: Auto-ins AutoInstallWbenOpenint): Auto-in:	ed Font stall when defini stall when open	YourTranslati Hinzufügen	on					*

Fig. A.2: The localization tool L10N

The menu bar comprises the menu items **File**, **Import** and **Export**. The main functions of the localization process can be accessed using the **File** menu. These are the commands **Accept Translation** (Strg+S), **Read System** and **Write and Compile Catalogs** (Umschalt+Strg+S).

The **Read System** command (it can also be selected by pressing the **Edit...** button in the **Settings** dialog box) Starts the search for constructs with user messages in all the methods of the application. The user messages are displayed in the **UserMessages** window.

The Accept Translation command confirms the Translation entered and writes it in the resident translation memory.

The **Write and Compile Catalogs** command writes the translations into the LBL files and the IDX files, so that they can be reused in other *VisualWorks* installations.

The File menu also comprises the menu items Options, Original Language, Empty TM and TM Stats.

After choosing the **Options** menu item, the **Settings** dialog box will be opened. In this dialog box, the settings for the localization tool can be changed (see Figure A.3). It comprises the check boxes **Remove translated entries from the UserMessage list**, **Skip translated entries in the UserMessage list**, **Include translation from Cnone when 'en' is the source** and **Use CaseSensitive comparison for existing translations**. Furthermore, it comprises the fields **Number of matches shown (Show all, 1, 3, 10, 25, 50, 100)**, **Source directory** and **Target directory**.



Fig. A.3: The Settings dialog box

In the **Penalties** dialog box (see Figure A.4), which can be accessed via the window at the left hand side of the **Settings** dialog box, you can adjust the penalties for the matches found in the translation memory. The penalties can be set from 0 per cent to 100 per cent, if the source text has another territory identifier than the source text of the match found in the memory, if the translation has no territory identifier, in contrast to the translation of the match found in the memory, or if the translation has another territory identifier than the translation found in the memory.

87

v3>Settings		<u>_ [□i x]</u>
V3>Settings   I Look and Feel   I - O Look and Feel   I - O Look and Feel   System   fc Fly-by Help   III' Loading   W Memory Policy   p Message Catalogs   (3 Parcel Path)   II' - O Source   II' Tools   • Oo Browser   • O Database   • Ä Debugger   B\[J MOn   I Painter   I W Workspace   Store   I Default Package   % Properties Publishing Polic	-S Penalties Source Language other country: Target language neutral country: Target language other country:	2% 2% A%
	rel 📔 🏠	Help

Fig. A.4: The **Penalties** dialog box

Using the menu item **Original Language**, you can display the user interface of an application either in the original language or in the target language. The menu item **Empty TM** can be used to empty the resident translation memory. With the menu item **TM Stats** you can display the number of entries currently stored in the resident translation memory (see Figure A.5).

	×
617023 Entries	are currently in the TM
Yund	OK

Fig. A.5: Translation memory statistics

The Import menu comprises the menu items Import Catalog, Import CSV Catalog, Import CSV Catalog Folder, Import TM, Import TMX and Import TBX. This menu can be used to import glossaries in the LBL file format or in the CSV file format, resident translation memories (formats: translation memory (.tm), Smalltalk file outs (.st), Smalltalk parcel sources (.pst), Smalltalk change files (.cha), Smalltalk parcels (.pcl), workspaces (.ws), text files) as well as TMX files (version 1.1) and TBX files.

The **Export** menu comprises the menu items **Export TM, Export TMX** and **Export TBX.** For the export, the same formats are supported as for the import.

In the upper part of the left hand third of the *L10N* window, you can select the source language and the target language, as well as the catalog to be translated. The display indicates how many translation units in the catalog are already translated. Further down is the **UserMessages** window with the translation units. Untranslated units are bold, translated units are italic. Further down is a window for the search for specific keys.

In the upper part of the right hand side of the *L10N* window, the methods containing the current translation unit are displayed. In the window to the right, the other translation units of the corresponding method are displayed. With the **Open in Class** button, you can open the class where the method is implemented (see Figure A.6).

#### Anhang



Fig. A.6: The class where the method is implemented

Further down is a window with the source coding of the method with the current translation unit. The methods created with the *VisualWorks Canvas* tool can also be displayed in a WYSIWYG mode (see Figure A.7).

Anhang



Fig. A.7: A method in the WYSIWYG mode

In the **Default String** window further down, the current source text is displayed. Further down is the **Translations** window with the current translation and any other former translation. Further information displayed in this window is the date and the time the translations were confirmed.

In the window further down, the translations found in the imported glossaries or translation memories are displayed. You can adjust the number of the matches shown (options: Show all, 1, 3,10, 25, 50,100) and the minimum quality of the matches (30 per cent up to 100 per cent).

Right at the bottom of the screen is a field where you can enter a new translation, as well as the **Accept Translation** button. With this button, you can confirm a translation and write it in the resident translation memory.

## 3 Evaluation

### 3.1 Introduction

*L10N* is a compact localization tool, which was especially designed for the localization of Smalltalk applications. It has the most important features of a localization tool, like the extraction of the text to be translated from the source coding, the WYSIWYG mode and the leverage function. *L10N* runs on several platforms like *Windows, Linux, Unix* and *PowerMac*. It Supports the translation memory and terminology exchange formats TMX and TBX, which guarantees an uncomplicated data exchange with other translation memory Systems and localization tools.

### 3.2 Advantages

The intuitive user interface makes the tool very user friendly. After a short settling-in period, the tool is easy to use, due to the fact that the tool's features and options are restricted to the essential part. There is a high conformity of the Steps performed by the System with the user's expectation.

The Status of the translation units and catalogs is quickly perceptible. Untranslated units in the **UserMessages** window are stressed by bold type, translated units by italic type. Furthermore, translated units can be extracted from the user messages list.

A specific feature *ofLION* is the possibility to switch between the original language and the target language within the WYSIWYG mode at all times during the translation process. This function is very useful to keep the consistency within the translation.

An example from the localization of *VisualWorks* is the **Smalltalk** menu in the **Workspace** window. (see Figures A.8 and A.9).

	Ctrl+D
⁄ Printit	Ctrl+P
风 Inspectit	Ctrl+Q
🕬 Debug it	Ctrl+B
<u>F</u> ile it In	
Narnespace	s

Fig. A.8: The Smalltalk menu in the Workspace window

\$ Ausführen	Ctrl+D
\$\ Printit	Ctrl+P
AH, Inspect it	Ctrl+Q
^* Debug it	Ctrl+B
Eile it In	
₿j <u>N</u> amespaces	- 

Fig. A.9: The Smalltalk menu in the WYSIWYG mode oiLWN, Option: display translation

In this case, the imperative form *Do it* has been translated with the infinitive form *Ausführen*. The sorting of the translation units in the **UserMessages** window is alphabetically and not in the order of their occurrence in the text. Therefore, the menus and dialog boxes can not be translated in a continuous flow. The possibility to switch between the source texts and the translations allows the translator to see, which translation strategy has been used so far, and to adopt it to the other translation units in the same menu or dialog box.

Another advantage *ofLION* is given by the manifold import and export possibilities. So, functions that are not implemented in *L10N*, can be executed with other programmes, for example the maintenance of the translation memory. Because a translation will be automatically written in the translation memory when confirming it, the translation memory usually contains also translation units that are not suitable for further use. The TMX or TBX export allows the data to be exchanged and manipulated with other translation memory or terminology Systems (see Figure A.10).

#### Anhang

ier I <noliteradiv.></noliteradiv.>		al 12	
i^Files		- 1 U&Dateien	J
And it SAA - Hander		siüber&VisualWorks	
3LA ocont		<sup>3</sup> Übernehmen	
Accept	<u>12</u>		
4 &Add		& HinzufUgen	
<sup>5</sup> Add &Changes		SAnderungen hinzufügen	
<sup>6</sup> &Add Class		<sup>6</sup> Klasse hinzufügen	
<sup>1</sup> Add directory		'Vérzaizhinis limizulMcjait,	
* &Add		<sup>a</sup> &Hinzufügen	
»(Addfile		<sup>3</sup> Datei binzufügen	
"IAdd8 Filo In	54	10 & File-In-Datei binzufügen	
"IAdd & Original Vargion			
	IS		
* &Add To			
<sup>13</sup> [Ad Hoc&SQL		«Ad-hoc-&SQL	
14 Align Bottom		<sup>14</sup> Unten ausrichten	
<sup>15</sup> Align Horizontal Center	er	<sup>15</sup> Horizontal mittig ausrichten	
«[Align Left		16II inks ausrichten	1
Importing file: DAdiptomarbeitMIQn\expo Σr ib nëtë kris * 1257 Import Comptele.	ort\TranslationsQ5Q7Q5 Imx		×
Field	I. Volue		
Greated	15.07.2005 22:29:23		1
Created By	Administrator		
E served de la construcción de la c	15.07.2005 22:29:23		
LastUsed	15.07.2005 22:29:23		-
Last Modified Modified Ev	Administrator		
Last Modified Modified Ey Context	Administrator TEXT		

Fig. A.10: Imported TMX file fromL70A^to SDLX

### 3.3 Suggestions for improvement

From the view of the translator, the subdivision into translation units by user messages is not the best solution, because the different meanings of homonyms can not be represented by different translations. Because only one translation per search key can be entered, a translation that reflects all the different meanings of the different concepts has to be found.

An example is the user message **Back.** On the one hand, this message can be found in the toolbar of a so-called **Inspector** and means *zurück* or *rückgängig machen* (see Figure A.11).

94



Fig. A.1 1: Elements of an **Inspector** toolbar (WYSIWYG mode in *L10N*)

On the other hand, the English user message **Back** also represents a menu item in the window menu of any *VisualWorks* window and means in this case as much as *nach hinten* (see Figure A.12).



Fig. A.12: Elements of a Windows menu

The alphabetical sorting of the translation units in the **UserMessages** window and therefore the processing sequence is not very practical. It would be better to have the possibility to sort and to translate the units accordingly to the order of their occurrence in the text. The translator could then translate all elements of a menu or a dialog box one after the other and would not have to get familiär with another context after each translation unit.

Because the search for translation units is only possible for their search key, but not for their description, not all translation units can be found. An example is the menu item **Class Methods**, which has the search key **ClassDefinitions**. With the entry *Methods*, this user message will not be found, although fuzzy match search is basically supported bei *L10N*. According to the assumption that the translator will not know all the search keys with their descriptions, the possibility to search for the descriptions would be an advantage.

The following improvement suggestions are not as significant as the ones already mentioned.

The **Default-String** window, which is used to display the entire current translation unit, should be adjustable, so that longer translation units can also be entirely displayed.

A check for double-time assigned hot keys is only possible in the WYSIWYG mode, that means, elements, that can not be displayed in the WYSIWYG mode, can not be checked. Therefore, a feature to check the hot keys for their uniqueness would be helpful.

The statistics created by the System only give information about the number of entries in the translation memory (including glossaries) and are therefore little informative concerning the leverage gained by a translation memory. A comparison of the translation units in the current project with the ones in the translation memory or in glossaries would be helpful to find out, whether it pays to inelüde the memory or the glossaries in the project.

There should be a possibility to display translation memories and glossaries that are included in the project and also a possibility to exclude them from the project.

Another improvement would be the possibility to include comments to the translation units, for example for translation alternatives.

### 3.4 Summary

*L10N* is a user-friendly localization tool, which is compatible with other tools for computeraided translation, due to the support of the standardised data exchange formats TMX and TBX. It has been especially developed for the localization of Smalltalk applications. For future versions of the tool, a solution for the treatment of homonyms should be found. It would also be an advantage if the sorting of the translation units could be chosen accordingly to their oecurrence in the text. Another improvement could be more extensive statistical information about the project and checking functions, for example for double-time assigned hot keys.

# Eidesstattliche Erklärung

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Hilfsmittel verwendet habe.

Köthen, 06.09.2005