




Localizability

Engineering Software for a Global Market





Advantages of Localized Software

- global use of software
 - increases productivity
 - competitive edge
 - people expect their software to “talk” to them in their language
 - legal issues
- 




Common Applications

Applications localized INTO English:

- Linux – Finland
 - Knoppix, SuSE – Germany
 - Mandrake – France
 - Opera – Norway
 - Nintendo – Japan
 - StarOffice (now OpenOffice) – Germany
 - ICQ – Israel
 - AVG Anti Virus – Czech Republic
- 




Different Steps of Localization

- **Localizability** – make a program localizable
 - extracting all language-dependant strings in binaries and putting them in a separate location that can be modified by a translator
 - **Translation** – into foreign language
 - strings, help, manuals, aux. files
 - **Localization** – country/culture-specific “codes”
 - moral values, work habits, social structure, esthetics, legal issues
 - **Internationalization** – make code language sensitive
 - string manipulation, system locales (decimal, date/time format, currency, address format)
 - **Globalization** – mostly political issues
 - maps, minority languages
- 




Preparing Code for Localization

- C++, C# (Visual C++)
 - string tables in resource files
 - translation using resource editor possible
 - Passolo, Catalyst
 - allows to resize/move GUI elements
 - Java (NetBeans, Eclipse et al.)
 - property file/resource bundle depending on system locales
 - XML based (REALbasic et al.)
 - common export/import format
- 



Common Problems

- GUI
 - most languages longer than English, text does not fit anymore
 - string concatenation
 - text only seen out of context, second part unknown eg. “of” or “to”
 - sentence fragments (one word “sentences”)
 - issues for languages with word gender (edit, “none” - of what?)
 - use of placeholders
 - grammar and word order are different
 - issues for languages with word gender (gender of word unknown)
- 

Common Problems...

(cont.)

- localized Windows versions
 - paths and folder names may be different (“My Documents”, “Program Files” can be localized)
- number formats
 - decimal (point vs. comma)
 - leading zero may be required
- regional options
 - date, time, address, phone number, currency
 - requiring state for countries without states
 - order of address is different
 - variable number of zip code digits

Common Problems...

(cont.)

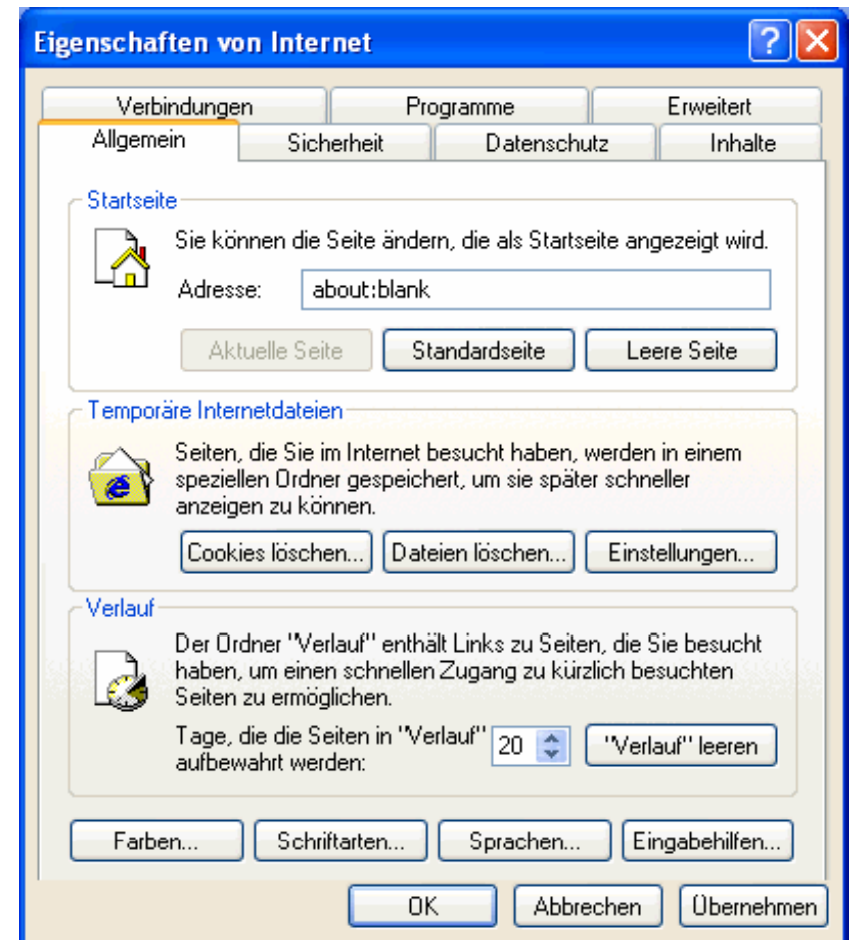
- external compatibility
 - database (csv files), formats
 - device interface
- parts of application “untranslatable”
 - text on graphics or icons
 - custom modules
 - “symbolic” does not “translate” (clipboard, color key)
- incompatibilities with character sets
 - application uses ASCII text instead of Unicode
 - double-byte languages
 - bidirectional text
 - accented characters

Comparison of GUI Elements

English

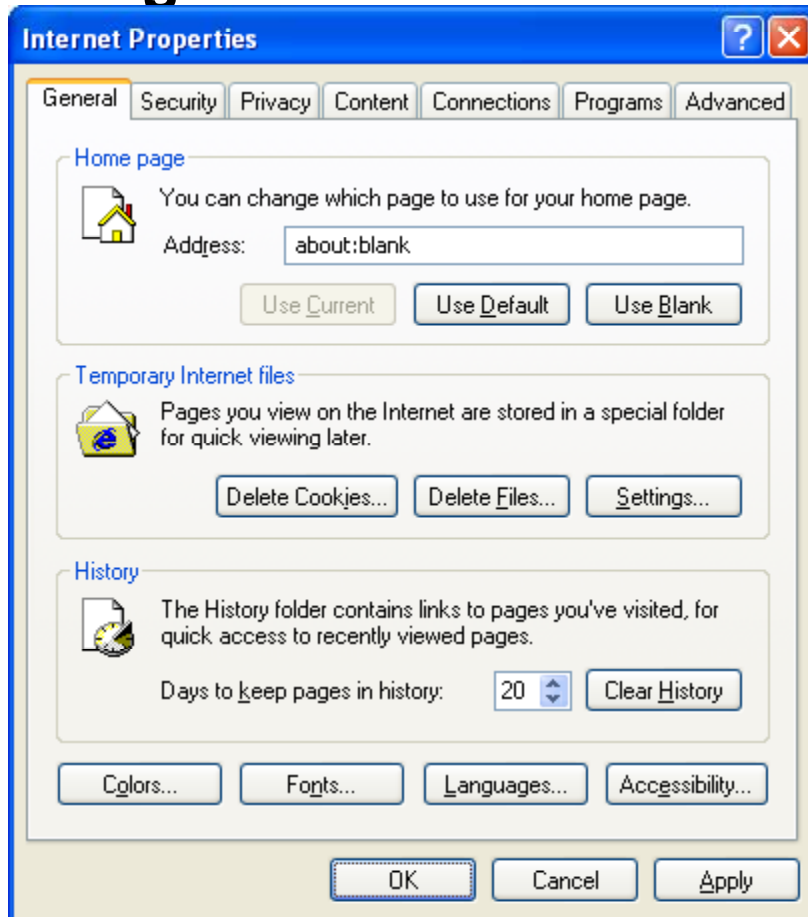


German

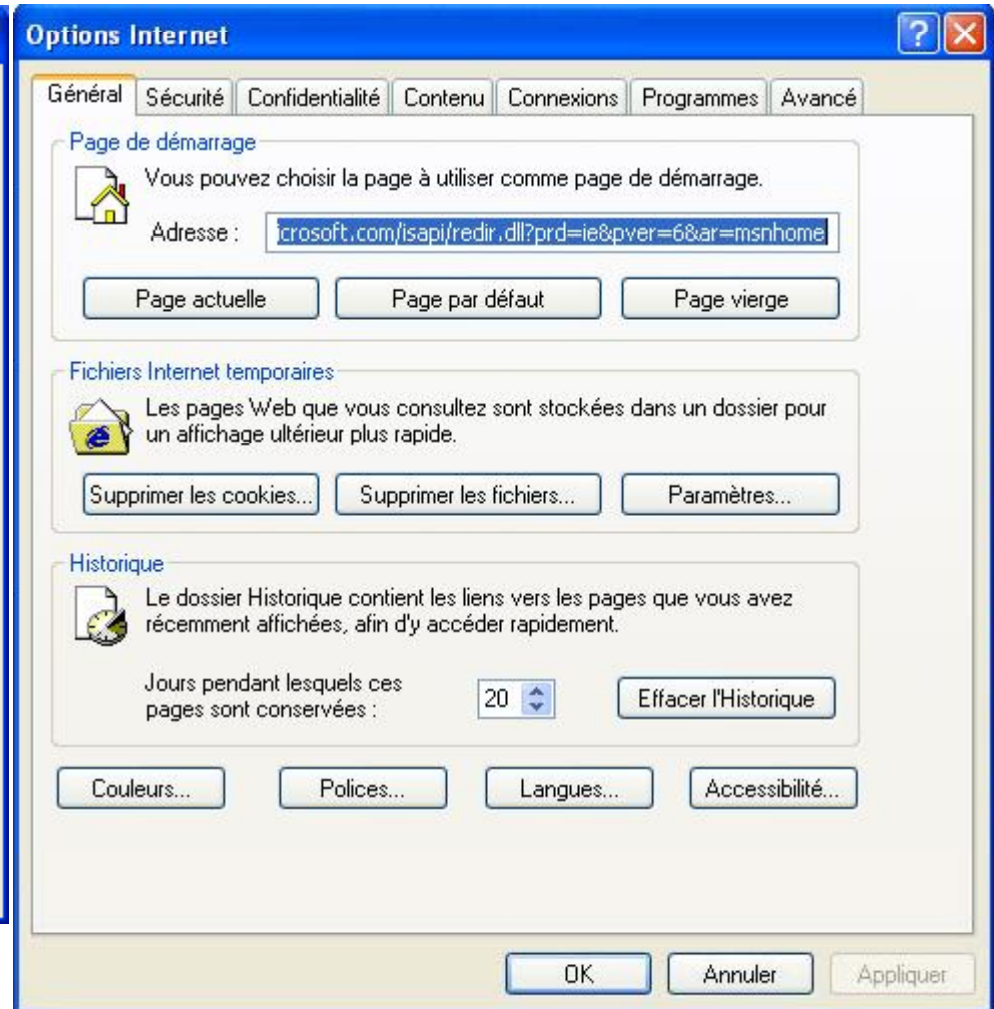


Comparison of GUI Elements (cont.)

English

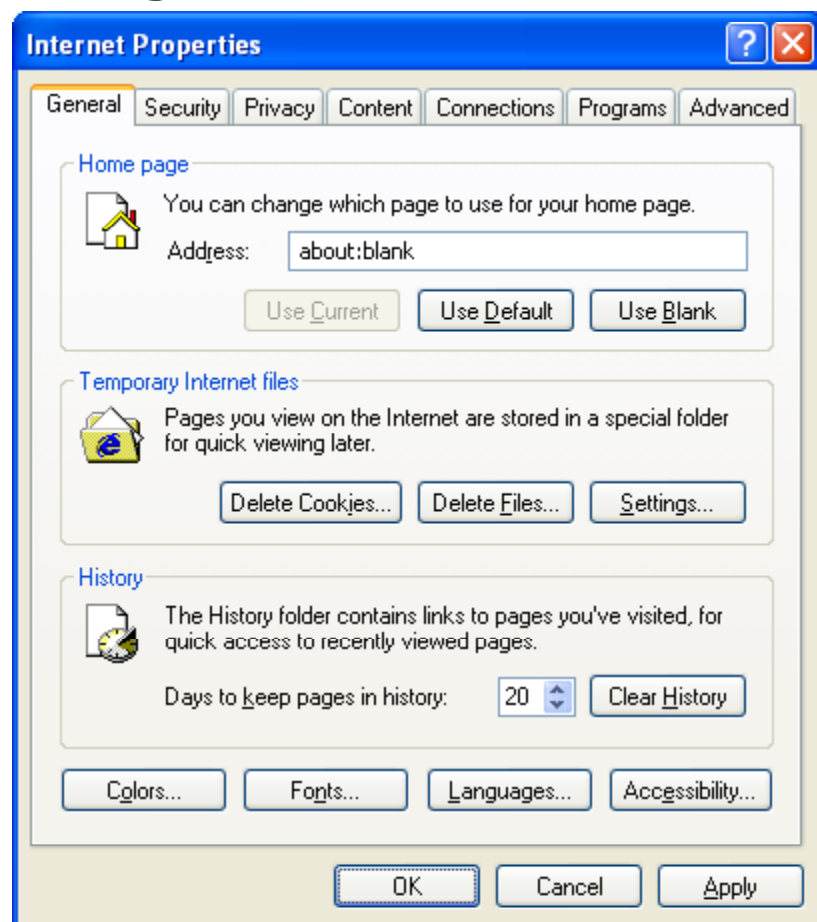


French

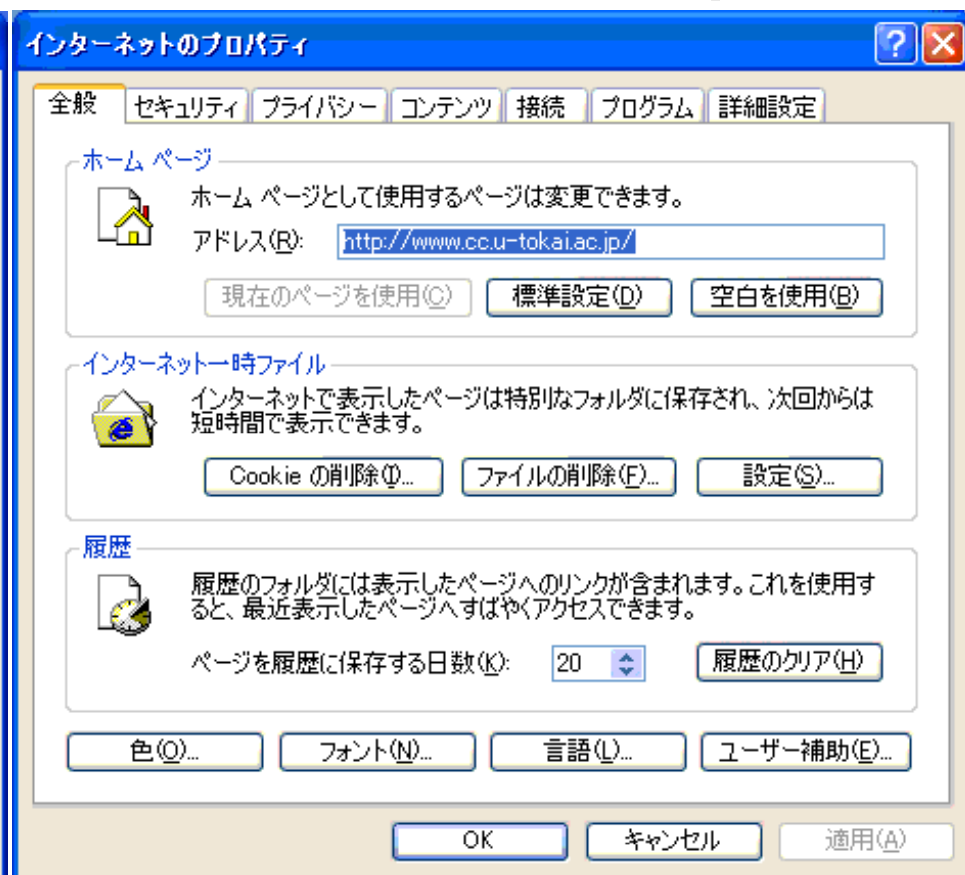


Comparison of GUI Elements (cont.)

English




Japanese





Affect on Project Release

- failing to prepare for localization can be extremely expensive and time consuming
 - application does not even start (Asian languages)
 - accented characters are lost or lose accent
 - parts of the application remain untranslated
 - string truncations within the GUI
 - extra testing time needs to be scheduled
 - tested and finalized English code must be changed and retested
 - ⇒ cost and people-hours increase
 - ⇒ deadline cannot be met
 - ⇒ project goes over budget
- 




Why does this Happen?

- application was not meant to be localized
- lack of localization knowledge/sensitivity
 - translator was not involved
 - foreign languages are treated as inferior releases
- time constraints
 - “I’ll fix that later”
 - “We don’t have time for that”
 - translators have to start working on a half finished product



Create a Resource Bundle in Java

- instructions for string extraction
 - NetBeans
 - Tools/Internationalize/Internationalization Wizard
 - Eclipse
 - Source/Find Strings to Externalize
 - strings are extracted into text file which can be translated
 - no GUI modification possible for translator
 - requires additional work and testing
- 



More Information

■ General

- <http://www.lisa.org/>
Lisa – The Localization Industry Standards Association
- <http://www accurapid.com/journal/softloc.htm>
Per N. Dohler – Facets of Software Localization

■ Java

- <http://java.sun.com/javase/technologies/core/basic/intl/faq.jsp>
Java Internationalization FAQ

■ CAT (Computer Aided Translation) and other Tools

- <http://www.passolo.com/en/home.htm>
Passolo – Resource Editor
 - <http://www.sdl.com/en/products/translation-memory/sdl-trados/>
SDL Trados – Translation Software
- 