Tech Info Library

WorldScript: Difficulties of Multilingual Software Development

Revised: 1/11/93
Security: Everyone

WorldScript: Difficulties of Multilingual Software Development

==========

Article Created: 6 January 1993

TOPIC ----------

This article describes the obstacles presented to users and developers of multilingual software before the introduction of WorldScript.

DISCUSSION ----------

Inconveniences to Users

For users worldwide, the situation before WorldScript created several disadvantages:

- Delayed delivery of new technology
  For users, the reengineering of the base system software into different languages meant a delay in the delivery of local applications, local system software, and hardware.

- Inconvenient or inefficient text-input methods
  Asian customers also require an easy-to-use and customizable input method. This ease of use can be achieved via in-line input, which integrates text entry and display within a document, avoiding the constant appearance of the bottom line window to enter text.

  Because in-line input is more intuitive, many developers integrate it into their products by bundling the input method with their application. However, in the past, the only way to provide the necessary seamless integration between input method and application was to develop the two together. In addition to increasing costs, this process also meant that every application had to have its own input method. Users would have to learn one input method for working with a Japanese word processing program and another for working with a Japanese spreadsheet.

- Limited product selection
  The delays and the difficulty inherent in localizing products meant that many developers simply didn't bother to enter some markets, limiting the availability of a variety of software and hardware products.

Obtstacles for Developers

For developers, the burden of developing on reengineered software bases raised some roadblocks:

- Time-consuming localization
  Even after the localized system software versions were available, developers had to spend significant amounts of time modifying the code in their application software to support other languages. They also needed to understand the code changes that were made in order to support a particular language in the system software. So, they couldn't apply the development of localized applications for one language to another language. There were almost 30 different system software versions.

- Implementation of features in different languages
  Certain features are difficult to implement in other languages. As an example, fonts for the Japanese market can occupy more than 7MB, compared with the 40K to 60K required for a typical Roman font. In order to enter, display, and print such a font with comparable performance, special modifications must be made to the base system software. In the case of TrueType for Japan, these special modifications were well integrated, but they weren't part of the base system software.

- Complicated upgrades
  Each upgrade requires an entire new cycle of translation. System upgrades or product revisions require developers to revise their in-line input method as well as their application to support more than 30 languages.

International Software Efforts at Apple

Before WorldScript, Apple faced two major international software challenges:

- Time-consuming localization
  When Apple introduced a new operating system, localization teams had to translate and adapt that system into more than 30 languages. This meant not only translating the menu names and file names, but also changing the base system software to support right-to-left text or contextual forms. Most previous system software releases required different base system software to support each language. Translations for non-Roman languages could take more than a year.

- Difficulty expanding markets
  This delay in development meant a delay in releasing both software and hardware products, and made it difficult for Apple to expand into new markets.

The Solution: WorldScript

WorldScript greatly reduces this international burden, bringing benefits to users, developers and Apple alike. With WorldScript, the same system-level support already available for Roman languages such as English, French, German, and Spanish is provided for non-Roman languages such as Japanese, Chinese, Arabic, and Thai.

All of the language support needs, including contextual formatting, two-byte support, and multidirectional text capabilities, are provided via WorldScript in a single system software version. With WorldScript, Apple can release "world-ready" operating systems that support most of the world's languages.

Tech Info Library Article Number:11251