


国際化 (インターナショナルリゼーション)



「国際化」の問題



「国際化」の問題



日本へ持って来る

アメリカ製コンピュータ

何が困るか？

1. ハード環境： 電源の電圧が違う？

The diagram illustrates the process of bringing a computer from the USA to Japan. It features a map of Japan on the left and a map of the USA on the right. A blue arrow points from the USA map to the Japan map, with a computer icon on the arrow. The text '日本へ持って来る' (Bring to Japan) is written below the arrow. Below the USA map, the text 'アメリカ製コンピュータ' (USA-made computer) is written. To the left of the Japan map, the text '何が困るか？' (What are the problems?) is written, followed by a list item: '1. ハード環境： 電源の電圧が違う？' (1. Hardware environment: Is the power voltage different?).

3

「国際化」の問題



日本へ持って来る

アメリカ製コンピュータ

何が困るか？

1. ハード環境： 電源の電圧が違う？
2. ソフト環境： メッセージ・メニューが英語（コマンドも英語）

The diagram is identical to the one on slide 3, showing a computer being brought from the USA to Japan. However, the list of problems is updated to include a second item: '2. ソフト環境： メッセージ・メニューが英語（コマンドも英語）' (2. Software environment: Messages and menus are in English (commands are also in English)).

4

「国際化」の問題



何が困るか？

アメリカ製コンピュータ

1. ハード環境： 電源の電圧が違う？
2. ソフト環境： メッセージ・メニューが英語（コマンドも英語）
3. ソフトで扱うデータが英語のみ
4. 出力（ディスプレイ・プリンタ）や入力（キーボード）が英語のみ

ここでは
「国際化」（多言語対応）
を考えます

多言語化へのアプローチ

- アプリ・ソフトから見て、なるべく手間が少なく
 - アプリを言語ごとに作り直すのはいやだ
簡単な設定で、表示・入力、メッセージが対応してほしい
 - OS自体も、多言語の対応が容易な方がいい
- (最近のもう1つの主張点) :
複数の言語の混用 (同時表示)
 - 例 : 同一ワープロ中・ブラウザ中に言語混在



国際化とは (教10.4)

- 国際化 (i18n = internationalization の略) :
 - 多言語・多様性を共通に処理できる仕組・枠組み
 - 1つのシステムで多様な言語・文化に対応させたい
- 地域化 (localization)
 - その地域に合わせること
- たとえば、OSとかソフトとかで
 - 国際化 : 設定を変えれば英語版にも日本語版にもなる
 - 地域化 : 設定のための情報やサブシステムを提供する
日本語表示・日本語入力、日本語メッセージ



ロケールとは

- 様々な国や言語の指定
 - 国際化の枠組みの中で、文字だけでなく、通貨単位や日付の書き方、文字のソート順序などの組み合わせで指定する
- 例：POSIXのロケール指定
 - LC_MESSAGES 言語と、メッセージの言語を指定
 - LC_CTYPE 文字のクラス
 - LC_COLLATE 文字のソート順序
 - 英米では A,B,C,⋯,a,b,c,⋯ 仏では A,a,B,b,C,c,⋯
 - 次ページへ続く



9

ロケールとは

- 例：POSIXのロケール指定 続き
 - LC_MONETARY 通貨記号や通貨の単位・書き方
 - LC_NUMERIC 数値の書き方
 - 米 ピリオドが小数点、コンマが整数3桁ごと
 - 独 ピリオドとコンマが逆
 - ヒンディ 整数が3桁より上は2桁ごとにコンマ
 - 65,23,14,159.278
 - LC_PAPER 紙のサイズ
 - LC_NAME 名前の順序
 - その他、住所・電話番号・測定単位など



10

文字コードについて

- 文字コードとは
 - 文字をコンピュータ内で表すための数値
例：Aは 41_{16} 、Bは 42_{16} 、aは 61_{16} 、など
 - 例：ASCII（教科書表10.2）
英数字＋記号で128文字分を7ビットで表す
- ことば
 - 文字コード系 文字の表現体系
 - 文字集合 文字コード系で表せる文字の集合
 - コードポイント それぞれの文字に対応する数値

11



いろいろな文字コード

- 英文(英数字・記号)
 - ASCII PC, Unix/Linux, ...
 - EBCDIC IBM (昔のメインフレーム)
 - ヨーロッパ語
- その他の様々な言語
 - 8ビットで表せる文字体系 vs
8ビットで足りない文字体系 ~ 漢字など
- 同一文書に複数言語を混在させたいとき
 - 更にビットが必要 Unicode ← 現在の方向

12



(脱線) いろいろな漢字コード

- 漢字コードは複数あり、文字化けのもと
 - JIS漢字コード 第1水準・第2水準の漢字にコード
⇒ これは原則唯一のコード表(2バイト)なのだが...
 - 英数字 (ASCIIやEBCDIC) と重なる部分があるので
漢字と英数字が混在できるように
適当にずらして、同じコード表に組み込む
⇒ ずらし方が複数できてしまった
- ポイント：標準化のタイミング
 - 2バイトコードの標準化のタイミングを失した(?)
⇒ 製品普及に間に合わず、複数の体系が並立

13



東邦大学

(脱線) いろいろな漢字コード(続)

- (7ビット)JIS漢字コードをそのまま用いる
 - 英数字との混在は切替符号(シフトアウト・イン)で
 - ネットではメール本文がこのコードを使っている
- Shift-JIS
 - ASCII + 8ビット目が1のところを漢字に使う
 - マイクロソフト MSDOS, Windowsで使われている
- EUC (Extended Unix Code)
 - ASCII+8ビット目1を漢字に/Shift-JISとは違う方法
 - UNIXで使われていた (最近ではUTF-8に移りつつある)
- Unicode/UTF-8 標準の本命

14

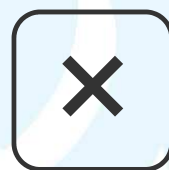


東邦大学

ここまでのまとめ

1. 国際化・地域化により、容易に多国で使うための枠組みが定義されている
 - 言語・文字だけでなく数や日付の書き方なども違う
それほどに、国・文化の違いが大きい
 - ロケールは地域選択のための記述子の例
 - (枠組みが無い頃は英語版を苦勞して作り替えていた)
2. 漢字コード体系の混乱・混在は苦い歴史
 - このようなことが起こらないように気をつけよう

国際化の考え方が
理解できましたか？



↓
次へ