Localization
and
Coded Character Sets

Internationalization/Localization

ISO/IEC 10646 (Unicode)

OCT.-2004 Bangkok, Thailand

Takayuki K. Sato

Chief Researcher, CICC-Japan

Objective of Localization

Is NOT writing a program that is capable to I/O local character set.

Is to provide a tool to bridge a digital divide

Is to make a country competitive

Bridging Digital Divide

- Use the world class software
- In local custom and language
- with interoperability within world network
- with the compatibility of data
- at the same price and availability

Reinvented equivalent products does not meet the objective

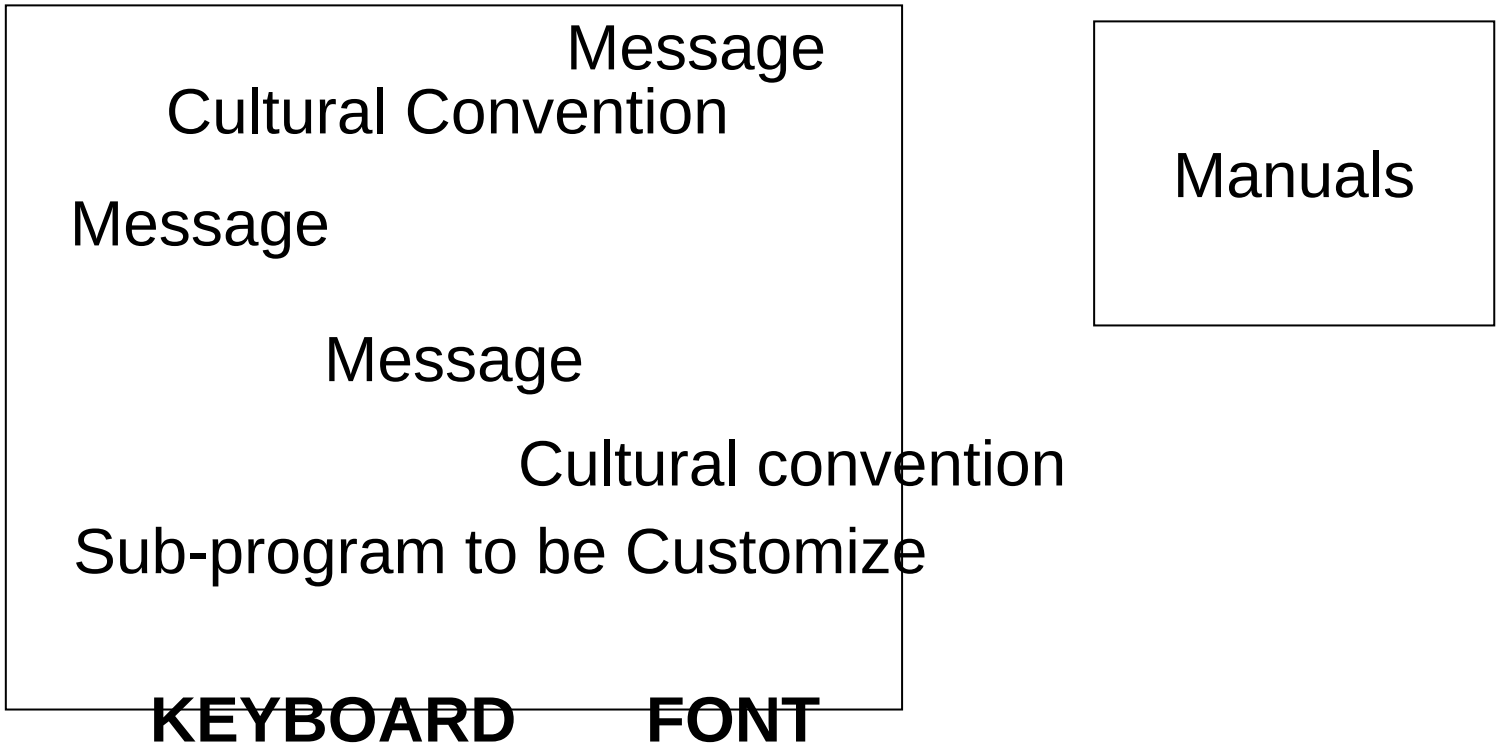
Local User Requests

following in local custom and language

- Operating Manuals
- Out put in local character (character handling)
- Friendly Input
- Cultural Conventions
- User Interface
- Data Processing in local custom
- Custom Functions
- Updates in Timely and free of charge

Meeting all above should be called localization

Old Style Hard Code Model



Costly and Time Consuming Localization

Hard Coded Model is “Search, Modify and Test”

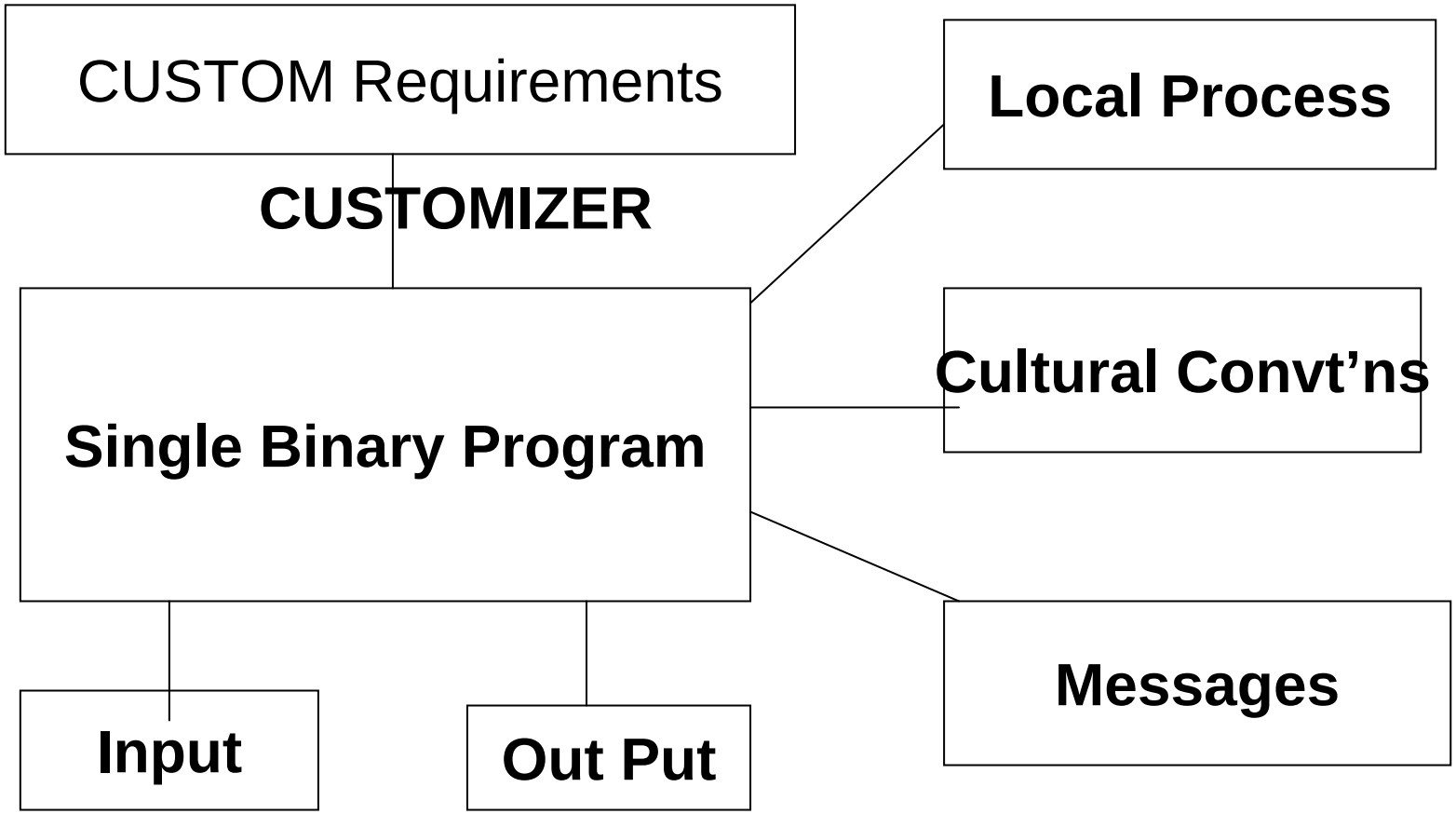
every pieces are costly and time consuming

Modification causes incompatibility and Quality Problems

Maintenance Issue

Maintenance should be done by local including backward compatibility

I18N/L10N model



All Cultural Dependents should be outside main program

“Single Binary” is a way to go

- Do NOT include any cultural dependent items within a program
- Do NOT include any character set dependency within a program
- All cultural dependencies should be at separated places from a program

Single Binary

Is NOT available at free of cost

Is available only when there is NO character set dependency

Any local character requires Character set dependency, if unconditionally coded

Key elements of Coded Character Set

- Coding Scheme (container)
- Coded elements (characters on code)
- Behavior of the coded elements
- Coding order
- Naming of coded elements

Binary code is able to share (Single Binary) only when….

- Coding scheme is the same
- Behavior of coded elements are the same
- The same elements with different behavior can not share the binary code

ISO/IEC 2022 approach does not meet for Single Binary requirement

- Locking Shift is difficult to process (traditional concern)
- Code developers have a free hands to select the behavior (multi-behavior processing needed)
- No way to predict new behavior

ISO/IEC 10646 approach

- Wide coding Space no locking shift
- Common Behavior

  Compromise on coding is needed
  This is very unpopular for many people

Common Behavior

- Unification
- Character and Glyph separation
- Normalization, Combining Sequence

  Ambiguity Possibility Issue
  Dual encoding issue

Ambiguity of sequence

There are three different characters

木　林　森

木 木 木

木木木？林木？木林？森？

木 NSNJ **木木**

木 林

What is industry standard?

It is a choice,

It is not a truth

Answers for most of the objection are

“So What?”

Conclusions

- For cost effective localization, a Single binary is the way to go
- To make the single binary, character code should make compromise to fit to program
- Fonts and Input Methods should absorb the compromised results for user friendliness
- International standard is needed to disclose the local requirements