

UNICODE Table based Input Method (UNIT)

UNICODE Table based Input Method (UNIT)

Sriram Swaminathan
sriram.swaminathan@sun.com

Hideki Hiura
hideki.hiura@sun.com

Steve Swales
steve.swales@sun.com

Sun Microsystems, Inc .,

Introduction:

A new generation of input method framework called “Internet/Intranet Input Method Framework” (IIIMF) has been developed and deployed to allow, for the first time, a true multi-lingual approach to text input.

An Input Method is a piece of technology by which an application directs the User to type, select, and send text to an application.

While many input methods are complex and must be custom built for a given language, there exists a broad class of languages which can be input using a generic table lookup approach.

Unicode Table based Input Method (UNIT) has been developed as part of IIIMF to enable multi-lingual text input for this broad class of languages.

Target Audience:

Localization Engineers, Developers, Font designers.

What will you get out of this presentation ?

At the end of this presentation you'll be able to add your own Input Methods (as part of UNIT) to IIIMF for broad class of languages.

What is UNIT ?

As part of the IIIMF, UNIT is a server side generic, multilingual composition engine.

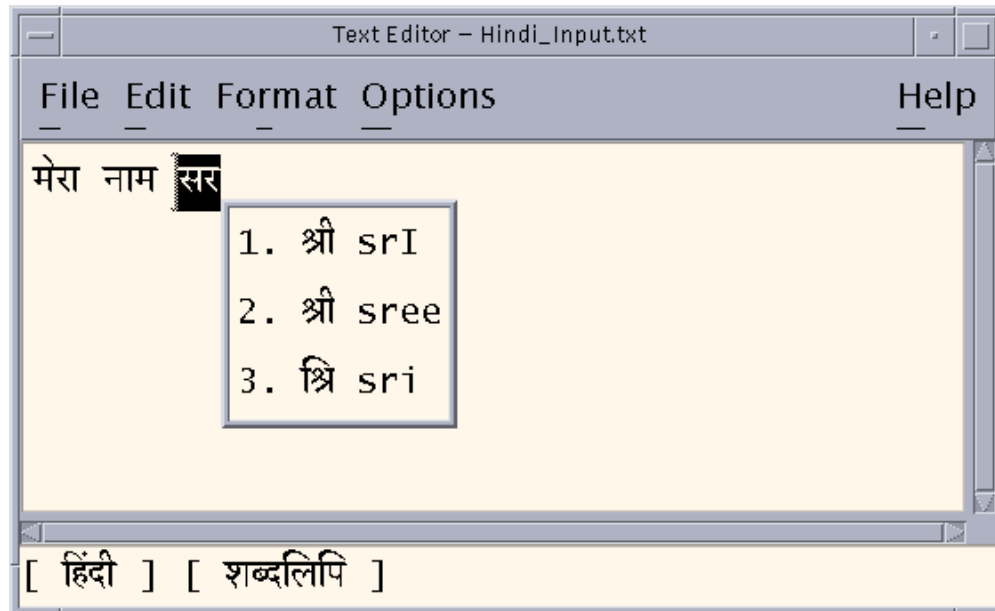
- The current UNIT engine supports :
 - * 8 Indic languages (Hindi, Bengali, Gujarati, Malayalam, Gurmukhi, Kannada, Tamil and Telugu),
 - * European, Cyrillic, Hebrew, Arabic, Greek and vietnamese
 - * Unicode Codepoint based input method (Unicode-Hex and Unicode-Octal),
- UNIT engine has the capability to support multiple keyboard layout (or) input methods for each of the supported languages.
- UNIT has the capability to add/remove supported languages as well as keyboard layouts using configuration file called “sysime.cfg”

How it works ?

Like other input methods, UNIT also provides features such as the following:

- A pre-edit region, which displays characters as the user enters them but before the user commits them
- A lookup choice region, which displays a list of choices and allows the user to select one
- A status region, which provides information such as whether conversion is activated and the current input mode or keyboard layout of the input method.

Figure-1 below shows the input method regions of UNIT, while trying to Input Hindi text.



Up to the point illustrated in Figure-1, the user has typed the keys 'sr' and is presented with a list of choices. The user can now commit one of these choices either by selecting the index (1 or 2 or 3), or by typing 'I' or 'i' or 'ee'.

How to disable/enable language/keyboard layout in UNIT ?

To enable/disable an Input language (or) keyboard layout, UNIT makes use of a configuration file called “**sysime.cfg**” located under \$(IIIMF_HOME)/locale/UNIT

The Input method specification file contains the following sections:

[GENERIC_IM_TABLE]

This section means that this is a generic codetable (based on UTF-8). No items are maintained under this section.

[SWITCH_LOCALE]

<Keycode Value> <Modifier Value>

After switching to UNIT Input Method, we can choose one of the available languages by typing the keycode & modifier value given above. Keycode and Modifier takes the value of IIIMF Keycode (not the X Keycode)

[SWITCH_LAYOUT]

<Keycode Value> <Modifier Value>

After selecting the preferred language, we can choose one of the available keyboard layouts (or) UNIT's submodules by typing values given above.

Note: IIIMF keycode values are provided (for reference) at the end of this presentation.

[Locale_Name_1]

<keyboard_layout_1> <Engine_Path> <LANG_DIR_1>
<keyboard_layout_2> <Engine_Path> <LANG_DIR_1>
.....

[Locale_Name_2]

<keyboard_layout_1> <Engine_Path> <LANG_DIR_2>
.....

The last two sections given above, means that the Language Engine (UNIT) will look for the data files under :

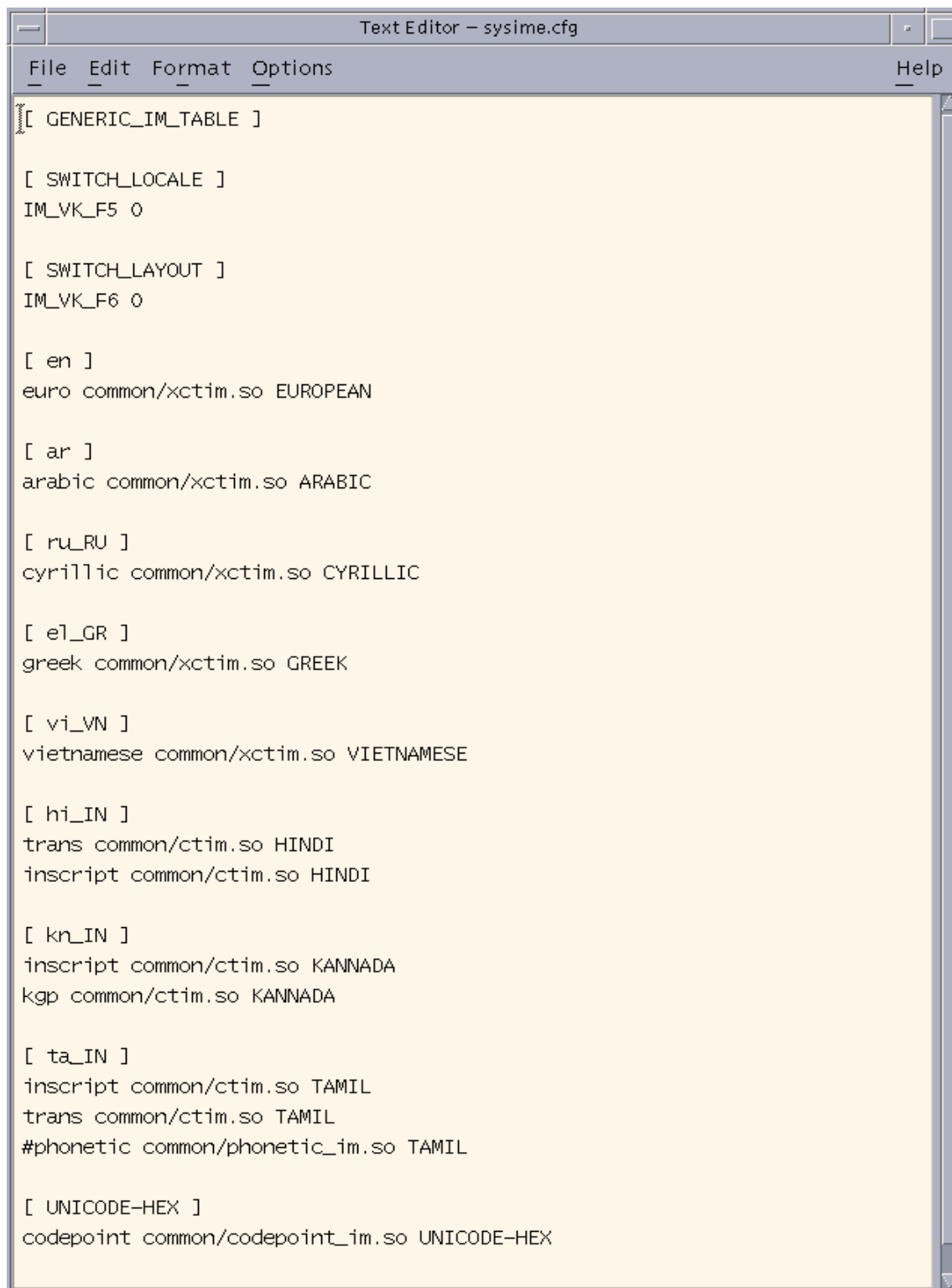
\$(IIIMF_HOME)/locale/UNIT/<LANG_DIR_1>/data/keyboard_layout_1.data
\$(IIIMF_HOME)/locale/UNIT/<Lang_DIR_1>/data/keyboard_layout_2.data
\$(IIIMF_HOME)/locale/UNIT/<Lang_DIR_2>/data/keyboard_layout_1.data

Engine_Path:

This is the location of UNIT's submodules such as codetable IM,codepoint IM ..etc. These submodules will be located under \$(IIIMF_HOME)/locale/UNIT/common

Unicode Table based Input Method (UNIT)

Figure-2 below shows a sample configuration file.



```
Text Editor - sysime.cfg
File Edit Format Options Help
[[ GENERIC_IM_TABLE ]

[ SWITCH_LOCALE ]
IM_VK_F5 0

[ SWITCH_LAYOUT ]
IM_VK_F6 0

[ en ]
euro common/xctim.so EUROPEAN

[ ar ]
arabic common/xctim.so ARABIC

[ ru_RU ]
cyrillic common/xctim.so CYRILLIC

[ el_GR ]
greek common/xctim.so GREEK

[ vi_VN ]
vietnamese common/xctim.so VIETNAMESE

[ hi_IN ]
trans common/ctim.so HINDI
inscript common/ctim.so HINDI

[ kn_IN ]
inscript common/ctim.so KANNADA
kgp common/ctim.so KANNADA

[ ta_IN ]
inscript common/ctim.so TAMIL
trans common/ctim.so TAMIL
#phonetic common/phonetic_im.so TAMIL

[ UNICODE-HEX ]
codepoint common/codepoint_im.so UNICODE-HEX
```

Input Method submodules under UNIT:

There are two submodules available under UNIT, the **CTIM (codetable based Input Method)** which is discussed in detail below and **Codepoint based Input Method** which when selected converts the HEX/OCTAL input (unicode codepoints) into UTF-8 characters.

How to add your Input Methods as part of IIIMF Standard Language Engine Interface LEIF ?

Depending on the requirement you can add your input method in the following ways :

1. The current UNIT implementation supports close to 15 languages. If you want to add an additional keyboard layout support for one of the UNIT's supported languages, then you can make use of **CTIM** (codetable based Input Method) which is a submodule of UNIT.

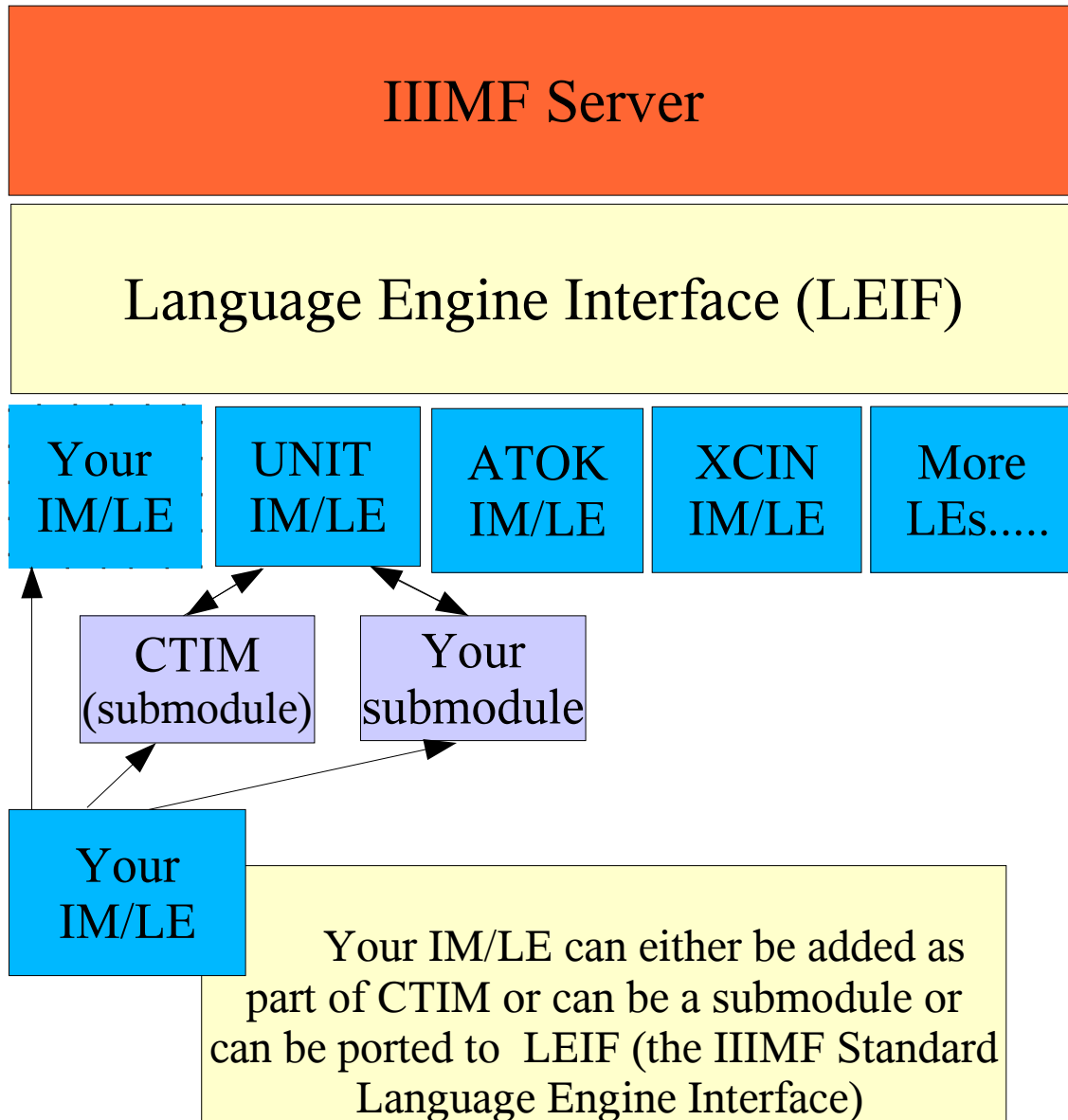
Note: **CTIM** is also delivered as part of UNIT.

2. If your requirement is to write an intelligent input method, then you can write your own submodule (such as **CTIM**) and install it under
\$(IIIMF_HOME)/locale/UNIT/common
and add an entry in the input method specification file under
\$(IIIMF_HOME)/locale/UNIT/sysime.cfg

Note: This saves lots of time for the Input Method developer, because of its simplicity. As UNIT language engine takes care of the communication with IIIM Server, your part would be process the input keyevent and send the output buffer to the client.

3. The third option would be to port your Input Method/ Language Engine to the IIIMF Standard Language Engine Interface LEIF.

Figure-3 below explains the above details.



CodeTable Input Method Interface

1. Creating a Codetable text file:

Codetable text file contains some function specific sections and a list of code-word mapping items. Few snapshots of the codetable text file are attached at the end of this document.

A codetable text file contains the following function specific sections:

[Description]
[Comment]
[Function_Key]
[Phrase]
[Single]
[Options]

Each section is briefly specified as below:

Section "[Description]":

This section describes some attributes of the codetable, such as encoding, name, valid characters, the maximum number of codes for one input items, and wild characters.

This section describes the following entry items:

1. "Locale Name:"
specify the locale name of this codetable (can be either Ascii or UTF-8 strings)
2. "Layout Name:"
specify the Keyboard Layout name of this codetable (can be either ASCII or UTF-8 strings)
3. "Encode:"
specify the encoding of this codetable (should be always UTF-8)
4. "WildChar:"
specify the wild characters for input codes (default values are '*' and '?')
5. "UsedCodes:"
specify the valid characters to input.
6. "MaxCodes:"
specify the maximum number of input codes for one item.

NOTE: The UTF-8/ASCII strings we maintain for "Locale Name" and "Layout Name" appears in the Status Window of each application that uses UNIT input method.

Section "[Comment]":

This section can be used to enter comments or information for explanation.

Section "[Function_Key]":

This section describes the key definition of some function keys, such as PageUP key to scroll up the candidate items, PageDown key to scroll down the candidate items, Backspace Key to delete an input code, and ClearAll key to cancel the input keys.

This section contains the following entry items:

1. "PageUp:"
2. "PageDown:"
3. "BackSpace:"
4. "ClearAll:"

Note: '^' means [**Control**] key, for example: '^N' means '[**Control+N**]' key.

Section "[Options]":

This section describes the options of the codetable input method, such as whether display help information for each candidate items, whether display the prompt string of the input key in preedit area, whether display the lookup candidates key by key or only display the lookup candidates when the Space key is pressed, whether commit the candidate when only one lookup result, and the select key mode: Number mode or Lower case mode or Upper case mode.

This section contains the following entry items:

1. "HelpInfo_Mode:" Values: "ON" or "OFF"
2. "KeyByKey_Mode:" Values: "ON" or "OFF"
3. "KeyPrompt_Mode:" Values: "ON" or "OFF"
4. "AutoSelect_Mode:" Values: "ON" or "OFF"
5. "SelectKey_Mode:" Values: "Number", "Lower" or "Upper"

Section "[Single]":

This section describes the input codes and their corresponding single UTF-8 characters. These characters must not be separated by a Space key.

The format of every line as follows:

keystroke_sequence Characterlist

Note: "CharacterList " means a list of UTF-8 characters with no Space separated.

Section "[Phrase]":

This section describes the input codes and its corresponding phrase words. these Indic phrase words must be separated by Space key. The format of every line as follows:

```
keystroke_sequence word1 word2 word3 ...
```

2. Convert the Codetable text file to binary format.

The utility tools "txt2bin" can be used to convert a text codetable file to binary file that the codetable input method interface can recognize.

The tool "txt2bin" is under directory:

```
$(IIIMF_HOME)/locale/unit/common/
```

The command syntax is:

```
# /usr/lib/im/locale/unit/common/txt2bin
   <source_codetable_file>
   <binary_codetable_file>
```

3. Installing the data file

Assuming you are adding a new keyboard layout for HEBREW, edit the \$(IIIMF_HOME)/locale/UNIT/sysime.cfg file as follows:

```
[ he ]
newdata <Engine_Path> <HEBREW>
```

Now copy the data file created in step 2, to

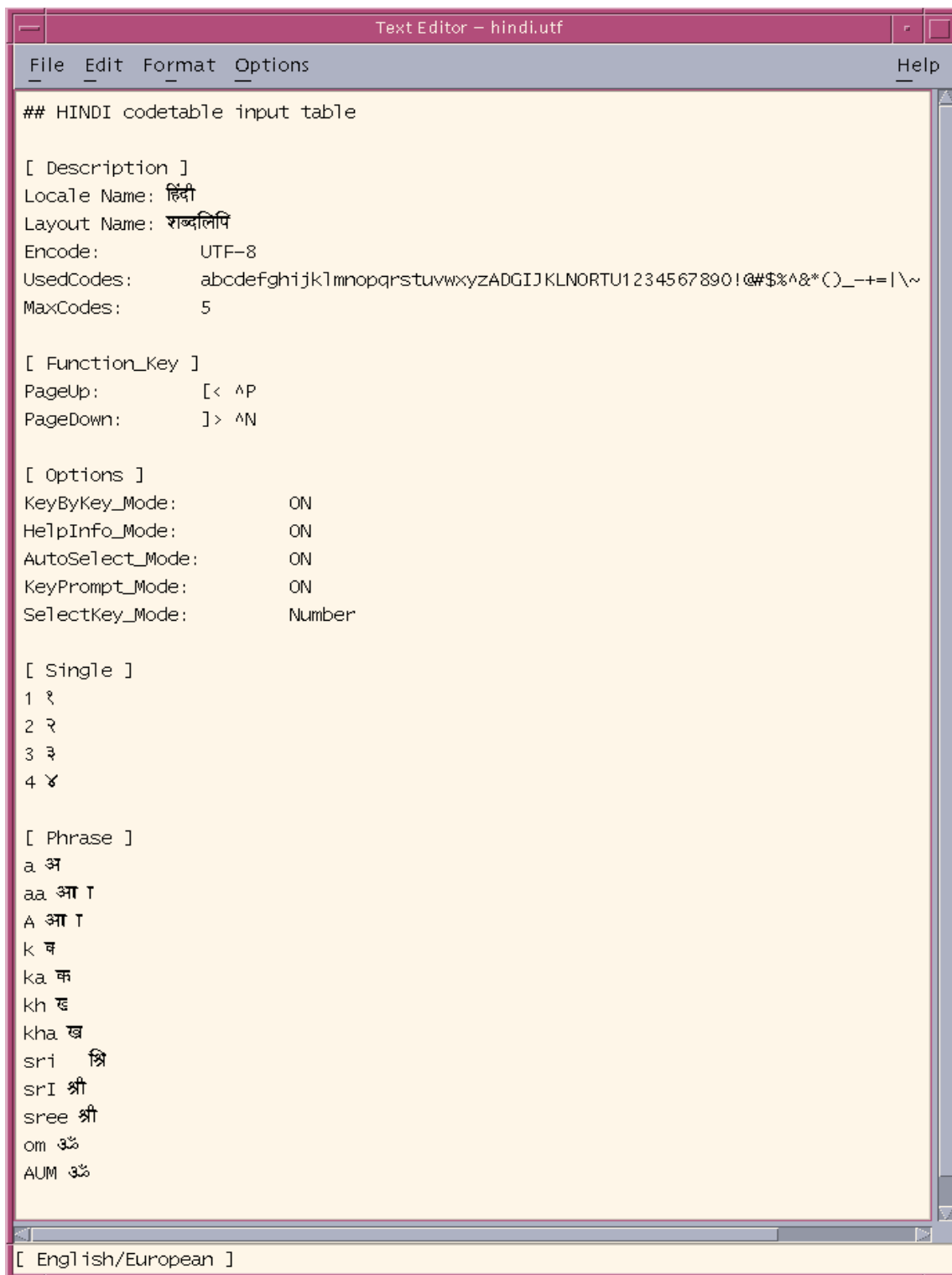
```
$(IIIMF_HOME)/locale/UNIT/HEBREW/data
```

4. Reloading the latest installed data files by IIIM Server

For the IIIM Server to reload the latest installed data files, send the SIGHUP signal to **htt_server** process.

Figures below are some of the sample codetable text files.

Unicode Table based Input Method (UNIT)



The image shows a screenshot of a text editor window titled "Text Editor - hindi.utf". The window has a menu bar with "File", "Edit", "Format", "Options", and "Help". The main text area contains the following configuration for a Hindi codetable input table:

```
## HINDI codetable input table

[ Description ]
Locale Name: हिंदी
Layout Name: शब्दलिपि
Encode: UTF-8
UsedCodes: abcdefghijklmnopqrstuvwxyzADGIJ KLNORTU1234567890!@#$%^&*()_+=|~\
MaxCodes: 5

[ Function_Key ]
PageUp: [< ^P
PageDown: ]> ^N

[ Options ]
KeyByKey_Mode: ON
HelpInfo_Mode: ON
AutoSelect_Mode: ON
KeyPrompt_Mode: ON
SelectKey_Mode: Number

[ Single ]
1 १
2 २
3 ३
4 ४

[ Phrase ]
a अ
aa आ I
A आ I
k क
ka क
kh ख
kha ख
sri श्री
srI श्री
sree श्री
om ॐ
AUM ॐ
```

At the bottom of the window, there is a status bar that reads "[English/European]".

Unicode Table based Input Method (UNIT)

```
Text Editor - greek.utf
File Edit Format Options Help
[ Description ]
Locale Name: GREEK
Layout Name: Greek
Encode:      UTF-8
UsedCodes:   abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ'-=\\[;'/./,<>
MaxCodes:    8

[ Function_Key ]
PageUp:      [< ^P
PageDown:    ]> ^N
BackSpace:   ^H ^?

[ Options ]
KeyByKey_Mode:      ON
HelpInfo_Mode:     OFF
AutoSelect_Mode:    ON
KeyPrompt_Mode:     ON
SelectKey_Mode:     Number

[ Phrase ]
516:4:  €
516:5:  €
65312:0:67:0:61:0:  €
65312:0:61:0:67:0:  €
87:0:   ς
69:0:   ε
82:0:   ρ
80:0:   π
2034:0: ς
2021:0: ε
2033:0: ρ
2036:0: τ
2037:0: υ
2024:0: θ
87:1:   Σ
65:0:   α
83:0:   σ

[ English/European ]
```

How to Install UNIT ?

As UNIT is part of IIIMF, building and installing IIIMF modules is necessary to make use of UNIT's capabilities.

You can download and install the latest IIIMF source/rpm from the following URL:

<http://www.openi18n.org/subgroups/im>

Conclusions :

Many input methods can be implemented using a generic table lookup algorithm. UNIT provides a framework for these input methods, as part of IIIMF. This allows developers to very easily add input methods to IIIMF for a broad class of languages. The examples and descriptions provided here should allow a developer to add new input methods to UNIT. UNIT also currently includes complete implementations for at least 15 languages.

References :

1. <http://www.openi18n.org/subgroups/im>
2. <http://www.unicode.org>