Documentación de **Transliterate**

Javier León Palomares

2 de diciembre de 2015

Índice

1. Introducción.

Transliterate es un programa que permite la transliteración bidireccional entre el alfabeto latino y el silabario japonés Hiragana (ひらがな). Además, puede analizar archivos HTML con un determinado formato en busca de bloques que transliterar. Para conseguirlo, hace uso de la herramienta *Flex*, que genera un analizador léxico con este propósito.

2. Requisitos.

Como requisito inicial, es imprescindible un sistema basado en Linux.

Para compilar **Transliterate** es necesario tener instalados *Flex* y *gcc*.

Para hacer uso de la opción de salida en formato PDF se necesitan los siguientes paquetes de LaTeX específicos:

- latex-cjk-common
- latex-cjk-japanese-wadalab
- fancyhdr
- color

3. Conceptos previos.

Para entender las funcionalidades que ofrece **Transliterate** es necesario aclarar el significado de algunos conceptos con los que trabaja.

3.1. Idioma japonés.

El idioma japonés está compuesto de dos silabarios y un conjunto de ideogramas de origen chino. Los silabarios son Hiragana (ひらがな) y Katakana (カタカナ), y representan la fonética de las palabras. El conjunto de ideogramas se denomina Kanji (漢字), y representan tanto fonética como conceptos.

3.2. Hiragana (ひらがな).

El silabario más utilizado en la escritura japonesa y el primero en aprenderse. Por ello, y tal como se ha mencionado anteriormente, **Transliterate** sólo trabaja con Hiragana.

3.3. Rōmaji (ローマ字).

Rōmaji (ローマ字) es el término utilizado en japonés para referirse al alfabeto latino. En Occidente se suele emplear para referirse al japonés escrito con letras latinas, en contraposición a la escritura original.

3.4. Transliteración.

La transliteración es el proceso de representar los símbolos de un sistema de escritura mediante los de otro. De esta forma, **Transliterate** consigue adaptar un texto a la forma que más nos interese.

3.5. Romanización.

Este término se refiere a la transliteración del japonés al alfabeto latino; no es un proceso estándar y cada método tiene sus ventajas y desventajas. La más utilizada y en la que se basa **Transliterate** es la romanización *Hepburn*; una tabla con las equivalencias se puede encontrar aquí.

4. Características.

4.1. Análisis de código HTML.

Transliterate puede analizar código HTML en busca de ciertas etiquetas que encapsulen Rōmaji, y es su acción por defecto. La implementación se ha especializado en la página web de letras de canciones AnimeLyrics.com por ser particularmente útil para el aprendizaje al poder combinar canciones y japonés. Gracias a esta especialización, **Transliterate** tiene además la capacidad de extraer el título de la canción e incluirlo junto con la transliteración en la salida que hayamos elegido.

4.2. Análisis de texto plano.

Transliterate tiene igualmente la capacidad de procesar directamente el texto deseado, omitiendo la fase de búsqueda.

4.3. Formatos de entrada y salida.

Los formatos de entrada admitidos son: archivo con código HTML, archivo de texto plano y texto por consola (para esto último, tras introducir el texto y presionar Enter se debe pulsar Ctrl + D para que el fin de archivo sea reconocido).

Los formatos de salida admitidos son: archivo de texto plano, PDF y texto por consola.

5. Limitaciones.

En primer lugar, hay que aclarar que las expresiones regulares no tienen la potencia para tratar con lenguajes naturales. Por este motivo, aunque reduzcamos el ámbito sobre el que trabajar, siempre habrá al menos algún matiz propio del poder expresivo de los lenguajes naturales que quede fuera del alcance de las reglas implementadas.

En segundo lugar, debido a la especialización del análisis de HTML implementado existe un límite en la aplicación de tal funcionalidad. Sin embargo, tratar de analizar un código HTML arbitrario mediante expresiones regulares es una tarea condenada al fracaso debido a sus diferencias dentro de la *jerarquía de Chomsky*; por ello, desde el principio sólo era posible procesar satisfactoriamente un subconjunto de posibilidades decidido por nosotros. Además, la variedad de formas de etiquetar los campos que buscamos por parte de los programadores es inabarcable.

En tercer lugar, **Transliterate** no implementa la posibilidad de transliterar Katakana o Kanji, el primero por decisión propia y el segundo por la complejidad y baja tasa de acierto que implicaría.

En cuarto lugar, al no haber un estándar universal para transformar los conjuntos de símbolos con los que se trabaja, en ciertas ocasiones habrá caracteres incorrectos. Esto ocurrirá, sobre todo, al transliterar de Rōmaji a Hiragana.

En quinto y último lugar tenemos una cuestión inevitable: podemos encontrar palabras o incluso oraciones en inglés dentro de un texto que se supone totalmente en Rōmaji. Si siempre estuvieran transformadas a fonética japonesa no habría problema, pero lamentablemente no es el caso. Esto ocurre sobre todo reconociendo canciones, donde los compositores tienen libertad creativa y los usuarios que añaden los textos a la web lo transliteran como ellos lo entienden. Al no disponer de un mecanismo para distinguir idiomas hasta este nivel, las palabras en inglés original cuyas letras coincidan con reglas serán también procesadas. **Transliterate** incluye una única convención para protegerlas y conservarlas en la salida, pero también es posible utilizarla para proteger Rōmaji; su uso queda a voluntad del usuario.

6. Uso.

Hay varios modos de ejecución para **Transliterate**, incluyendo uno por defecto sin argumentos.

- Sin argumentos: lee desde la entrada estándar un texto en HTML, translitera los bloques adecuados y los imprime en la salida estándar.
- Opción -i (**i**nput): cambia la entrada estándar por el archivo pasado como argumento.
- Opción -o (**o**uput): cambia la salida estándar por el archivo de texto plano pasado como argumento.
- Opción -p (**p**df output): cambia la salida estándar por el archivo PDF pasado como argumento.
- Opción -h (**h**iragana): omite la búsqueda de contenedores en HTML y translitera directamente el texto desde Rōmaji a Hiragana.
- Opción -r (**r**everse): realiza una transliteración inversa (Hiragana a Rōmaji).

Nota: Las opciones -o y -p son mutuamente excluyentes ya que se refieren a distintos formatos de salida.

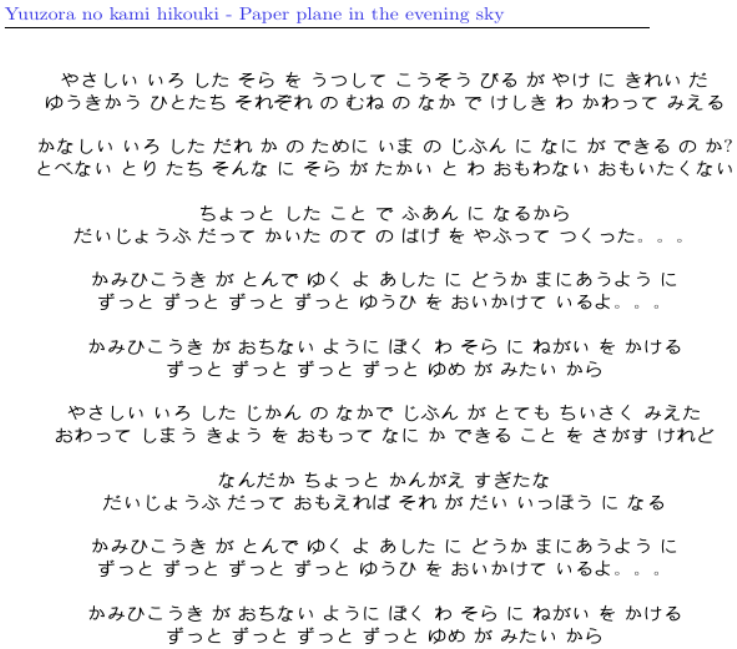
En caso de utilizar la opción -h en conjunción con -o y particularmente -p es conveniente tener en cuenta el nombre que se dará al archivo de salida, ya que se utilizará para generar un título.

En relación al formato del texto a transliterar, bastan tres consideraciones adicionales:

1. Las letras correspondientes a Rōmaji deben ser minúsculas, aunque se permite que la primera letra de la palabra sea mayúscula.
2. Como convención, las palabras (entiéndanse como agrupaciones de más de una letra) en mayúsculas se omitirán en la transliteración de Rōmaji a Hiragana. En su lugar, se transformarán todas sus letras excepto la primera en minúsculas y se imprimirán en la salida sin más cambios. Como apunte, es necesario decir que es únicamente una posibilidad puesta a disposición del usuario del programa de cara a la visualización del *output*; no es estándar en ningún sistema ni se debe esperar que otras personas cumplan esto en sus textos.
3. En la transliteración de Hiragana a Rōmaji se conservan todas las letras latinas sin necesidad de convenciones.

7. Ejemplo de uso.

A continuación tenemos una captura del resultado de una ejecución con las opciones -i para seleccionar un archivo como *input* y -p para obtener la salida en formato PDF.


Yuuzora no kami hikouki - Paper plane in the evening sky

やさしい いろ した そら を うつして こうそう びる が やけ に きれい だ
ゆうきかう ひとたち それぞれ の むね の なか で けしき わ かわって みえる

かなしい いろ した だれ か の ために いま の じぶん に なに が できる の か?
とべない とり たち そんな に そら が たかい と わ おもわない おもいたくない

ちょっと した こと で ふあん に なるから
だいじょうぶ だって かいた ので の ばけ を やふって つくった。。。

かみひこうき が とんで ゆく よ あした に どうか まにあうよう に
ずっと ずっと ずっと ずっと ゆうひ を おいかけて いるよ。。。

かみひこうき が おちない ように ぼく わ そら に ねがい を かける
ずっと ずっと ずっと ずっと ゆめ が みたい から

やさしい いろ した じかん の なかで じぶん が とても ちいさく みえた
おわって しまう きょう を おもって なに か できる こと を さがす けれど

なんだか ちょっと かんがえ すぎたな
だいじょうぶ だって おもえれば それ が だい いっぽう に なる

かみひこうき が とんで ゆく よ あした に どうか まにあうよう に
ずっと ずっと ずっと ずっと ゆうひ を おいかけて いるよ。。。

かみひこうき が おちない ように ぼく わ そら に ねがい を かける
ずっと ずっと ずっと ずっと ゆめ が みたい から

Figura 1: Fragmento del PDF resultado de la ejecución de **Transliterate**.