## Uso de InterText

### Ayuda

InterText cuenta con una ayuda contextual. Eso quiere decir que en la esquina superior derecha de la web normalmente veremos un link que pone “help” y nos llevará directamente a la sección de la ayuda relevante para la zona de InterText en la que estamos (gestor de textos, importación de alineado, edición de alineado...). Si estamos editando el alineado hay un botón (un icono de un libro con un interrogante).

El contenido de **dicha ayuda se encuentra aquí** por si la queremos tener toda a la vista:

<http://ecpc.xtrad.uji.es/Intertext/help.php>

## Trabajo por lotes

Incompleto

InterText cuenta con una serie de scripts alojados en la carpeta CLI (veáse el directorio donde está instalado InterText en el servidor o en el archivo comprimido que se puede descargar de la página de Pavel). CLI significa Command Line, es decir scripts de la línea de comandos (en este caso escritos en PHP) que sirven para realizar distintas operaciones relacionadas con InterText (importar, alinear, actualizar, exportar...) a través de la terminal de uno en uno o por lotes.

A continuación se describe cómo jmm ha utilizado los scripts CLI para procesar el material por lotes. Esto quiere decir que puede haber otras formas menos rudimentarias de usarlos, si alguien las encuentra que las comparta.

### IMPORT FILES (BATCH)

If working with Ubuntu server

#### Pre-processing

RENAME files

rename 's/(EN)(\d{8})(\.xml)/$2\.$1$3/' \*.xml

#### IMPORT VERSIONS

Go to /var/www/Intertext/cli

It might be a good idea to include the DTD in the same folder of XML files to avoid strange messages in the command-line

ENGLISH version

for i in ./intertext-source-en/\*.xml; do sudo ./import $i -e s ; done

#The folder intertext-source-en can be placed anywhere, just write the

Spanish version (files can be anywhere, just specify where)

for i in ./intertext-source-es/\*.xml; do sudo ./import $i -e s ; done

**IMPORT ALIGNMENTS (BATCH)**

**¡AVISO!** Este proceso es solo necesario si el alineado se ha realizado fuera de InterText (por ejemplo, usando la interfaz gráfica de TCA2). Puesto que nuestro InterText cuenta con TCA2 y HunAlign incrustados se puede alinear directamente (de uno en uno o por lotes) las versiones que se deseen. Además alinear directamente con InterText parece que evita problemas que nos han surgido al importar un archivo externo de alineado.

#### Preprocess ALIGNMENT

RENAME files

sudo rename 's/(EN)(\d{8})(\_)(ES)(\d{8})(\.xml)/$2\.$1$3$5\.$4$6/' \*.xml

RENAME attributes for file names

sudo perl -pi -e 's/(\p{Lu}+)(\d{8})(\.xml)/$2\.$1$3/' \*.xml

#### IMPORT ALIGNMENT

for i in ./intertext-alignment-en-es/\*.xml; do sudo ./align -i $i; done

Importing texts manually (using the web interface)

Call version files 20050111.EN.xml and 20050111.ES.xml

Change in the alignment file at least the atribute linkGrp toDoc instead of EN2005011.xml put 20050111.EN.xml and the same for the Spanish file.

And the name to 20050111.EN\_20050111.ES.xml

### ALIGN VERSIONS (BATCH)

### EXPORT FILES and ALIGNMENTS (BATCH)

Lo más fácil es, tras pasar todos los alineados a status “finished” de forma manual (para comprobar que todos los alineados están revisados y aprobados), pasar todos los documentos que vayamos a exportar al status “closed”.

#### Cambiar status a closed

Para hacerlo por lotes [se accede al phpMyAdmin del servidor](#id.m3a5txlxt6t) y en el lateral izquierdo seleccionamos la base de datos de Intertext. En la parte superior hacemos clic sobre la pestaña SQL y en el campo que aparece escribimos el siguiente comando que cambia todos los alineados de finished a closed:

UPDATE alignments SET status = '2' WHERE status = '1'

NOTA:

En la tabla alignments en la columna status hay varios valores posibles:

0 - open (se puede editar el alineado)

1 - finished (se puede establecer cuando todo el alineado ha sido revisado)

2 - closed (para indicar que ya está listo para exportar)

3 - blocked (para indicar a los demás que no toquen ese documento o simplemente un estado que no es ninguno de los anteriores).

Ejecutar script de exportación

A través de los scripts CLI de InterText se pueden ir exportando los textos y el alineado por lotes. Para facilitar las cosas se puede utilizar el script export\_batch.sh.

¿QUÉ HACE? Crear una carpeta para cada uno de los formatos en que se pueden exportar textos y alineados ([véase la ayuda de InterText](#id.72r1smh98ox6)), ejecuta el comando para exportar en cada uno de los formatos de exportación, y almacena en la carpeta correspondiente los documentos exportados.

¿DÓNDE ESTÁ? En el servidor virtual en la carpeta /home/administrador/InterText/intertext-export. Se colgará en SVN.

¿CÓMO SE USA? Situados en la carpeta donde se encuentra el script escribir en la terminal:

sudo sh ./export\_batch.sh

## InterText CLI MANUALS

A continuación está la ayuda de los distintos scripts distribuidos con InterText.

To see the help of any of this PHP scripts:

cd /var/www/Intertext/cli

php ./nameofthescript

To run any of this scripts:

cd /var/www/Intertext/cli

sudo php ./nameofthescript [and the options/parameters]

### IMPORT

InterText CLI import utility

============================

Tool to import text versions into the InterText system and import or generate alignments between them.

Usage:

import [options] <filename> [align <version2\_name> [alignment\_options]]

import [options] <filename> [align1 <version1\_name> [alignment\_options]]

Filename format:

<text\_name>.<version\_name>.<extensions>

Options:

-e <element\_names>

comma separated list of names of XML elements containing alignable text (e.g. '-e head,s,verse')

-n <text\_name>

text name (overrides the text name acquired from the filename)

-v <version\_name>

text version (overrides the version name acquired from filename)

Align:

Create alignment with version <version2\_name> or <version1\_name> after the import. The other version

must already be present in the system. Alignment is created by calling the associated 'align' utility.

Alignment options:

See the 'align' script for options.

Examples:

- to import text 'Document1' version 'english' from file 'Document1.english.xml':

import Document1.english.xml

- to import text 'Document1' version 'english' from file 'd1\_en.xml':

import -n Document1 -v english d1\_en.xml

- to import 'Document1' version 'french' from file 'Document1.french.xml' and create

automatic alignment with the version 'english' of the same document:

import Document1.french.xml align english

- to import 'Document1' version 'french' from the file 'Document1.french.xml' and create

alignment with version 'english', imported from file 'Document1.en-fr\_alignment.xml':

import Document1.french.xml align english -i Document1.en-fr\_alignment.xml

============================

### ALIGNMENT

InterText CLI alignment utility

===============================

Tool to import and/or generate alignment between two different versions of one text in the InterText system.

Usage:

align [options] <text\_name> <version1\_name> <version2\_name>

align -i <filename>

Options:

-i <filename>

import initial alignment from file

-swap

swap sides/versions when importing from file (option -i)

-d <status>

default status for imported alignment links with unknown status:

$statusdata (default: manual)

-m <method>[:<profile>]

specify method (and profile) for authomatic alignment, currently

available mathods: plain, tca2, hunalign (default: by settings.php, currently: '$DEFAULT\_METHOD:$DEFAULT\_PROFILE')

-r <user\_id>

set user-id of the user responsible for the alignment (default: '0' = nobody)

-e <user\_id>

set user-id of alignment editor (default: '0' = nobody)

-c\_chstruct

enable permission to change structure of the central text version (default: disabled)

-edit

enable permission to edit the texts (default: by settings.php, currently: '$DEFAULT\_EDIT\_PERMISSION')

-noedit

disable permission to edit the text (default: by settings.php, currently: '$DEFAULT\_EDIT\_PERMISSION')

-s <status>

set status for the alignment, possible values: open, finished, closed, blocked (default: open)

-v

verbose output (show progress)

===============================

### EXPORT

InterText CLI export utility

============================

Tool to export alignments and texts from the InterText system.

Usage:

export [options] aid <alignment\_id>

export [options] aname <text\_name> <version1\_name> <version2\_name>

export [options] t1aid <alignment\_id>

export [options] t2aid <alignment\_id>

export [options] tname <text\_name> <version\_name>

export [options] all\_closed

Options:

-f <format>

format of the output

for alignments:

xml - XML TEI file with links (plain IDs) [default]

xml:ic - XML TEI file with links (long IDs for project InterCorp)

xml:fn - XML TEI file with links (long filename-based IDs for project ECPC)

for texts:

xml - original XML file (plain IDs) [default]

xml:ic - original XML file (long IDs for project InterCorp)

xml:fn - original XML file (long filename-based IDs for project ECPC)

corresp - original XML file with added 'corresp' attributes on text elements,

linking the corresponding element IDs from the other version

(cannot be used in 'tname' mode)

segs - XML file with <seg>ments (ParaConc format)

for all\_closed:

-t <text\_name>

export only alignments of the given text

Examples:

- export alignment id 34

export aid 34

- export alignment of text 'Document1', versions 'english' and 'french'

export aname Document1 english french

- export second text version of alignment id 34

export t2aid 34

- export text version 'english' of 'Document1'

export tname Document1 english

- export all closed alignments with the texts into the current directory

export all\_closed

### RENUMBER

InterText CLI renumbering utility

==================================

Tool to renumber ID of the given text version in the InterText system.

Usage:

renumber [options] <text\_name> <version\_name>

Options:

-1

single level numbering only

===============================

### UPDATE

InterText CLI update utility

============================

Tool to update text versions already present in the database. The given file

will be compared with the database for each text element and in case of

difference the database elements will be updated with the new text. The

structure (and 'id' attributes) must be identical! If not, the script fails

as soon as some ID is not found in the database. However, changes commited up

to this point will NOT be revoked! (The changes are recorded in the changelog

table, at least.) Backup your texts before updating them!

The script will output a list of all changes done, including the ID and both

the old and the new text of every changed element.

Usage:

update [options] <filename>

Filename format:

<text\_name>.<version\_name>.<extensions>

Options:

-n <text\_name>

text name (overrides the text name acquired from the filename)

-v <version\_name>

text version (overrides the version name acquired from filename)

-u <user\_id>

set user-id as the author of updates (default: '0' = nobody)