Working with CTS

31. Juli 2017

1 Some basic SSH commands

Your data will be stored in Virtual Machine that represents a server. You can access and manipulate the files via the command-line interface

- log into the VM: ssh *[your VM'S IP]* -l *[your user name].* you will be then asked for your password
- transferring files into your VM: scp *[source directory of your file on your Computer] [your VM user name]@[your VM'S IP]:[the target directory on your VM]* you will be then asked for the password of your VM
- transferring folders into your VM: scp -r *[source directory of your file on your Computer] [your VM user name]@[your VM'S IP]:[the target directory on your VM]* you will be then asked for the password of your VM
- see which folder your in: pwd
- see content of the current folder: ls
- navigate into another folder (relatively from current folder): cd *[folder name]* or cd *[directory path]*

2 CTS ADMIN TOOL

Access the CTS Admin Tool via http://*[your VM'S IP]*:8080/cts_admin/

2.1 Setting up a new CTS instance

click 'create new CTS' and specify a name

2.2 Tab DB-Config

Important Parameters:

- **db_user:** username of your MySQL Login
- **db_pw:** password of your MySQL Login
- **db_name:** you have to choose a name (note that all your instances have to have a unique name)

2.3 Tab Data Import

Here you can specify how your data will be imported.
Important parameters:

- **resetDB**
  If set to true: all the datasets that were in de DB before are deleted.
  If set to false: you can add new datasets to the existing ones

- **sourceType** and **sourceDir**
  sourceType: 'cts' →sourceDir: give a URN to clone the documents from
  source Type: 'local' →sourceDir: give the source path on your VM where your dataset is stored

- **folderDelimiter** and **ignoreFolders**
  There are data sets (e.g. Perseus) in which the COLLECTION/WORK structure is not represented by the folder structure, but by the naming of the files. In this case you can mark which sign represents the folder in folderDelimiter. Then you also shoud set ignoreFolders to true.

- **docCount**
  Only important when you import data by cts cloning (sourceType='cts'). Here you give the number of documents you want to import.
  Example: If you type '2', only the first two documents of the instance you're cloning will be imported into your personal instance.

- **includeURNsWith** and **excludeURNsWith**
  Only important when you import data by cts cloning (sourceType='cts'). This works as a filter. you can type a substring here. Only URNs that include/do not include that substring will be added to your instance. The substring doesn't have to represent a complete element of the cts urn you want to import.
  Example: the WORK part of the source cts urn you want to clone has the following syntax: year.month.day. (e.g. '2001.10.16') includeURNsWith='2001' would import all documents of the source instance issued from 2001. includeURNsWith='200' would import all documents issued from 2000 on.
  You can combine filters using the ampersand '&'. Then all documents that contain *both* substrings are added.
  Example: '2001&10' would import all documents from October 2001, but also all instances that issue from the 10th day of every month in the year 2001.

- **sentenceSegmentation**
  They are datasets that are not marked up on a sentence level. If you set sentenceSegmentation=true, then the imported instance will be marked up on a sentence level.

- **createMissingN, ignoreNFromFile & MissingNStartWith**
  N refers to the attribute 'n' in the xml documents that are the source for the cts. n serves as identification for the elements in the PASSAGE part of the cts i.e. the elements that are below document level and for that reason not structured by a folder hierarchy. By identifying them using a numbering they can be referred to. Consecutive n's don't necessarily have to be consecutive numbers. You can rather think it as an ID.
  Select *createMissingN* if the structural elements of your document (e.g. paragraph and sentence elements) don't have an n attribute. Then this identifier will be added using increasing consecutive numbers. If you select *ignoreNFromFile*, the original n identifiers from the source document will be deleted. Enabling createMissingN will then add new consecutive n identifiers.

2.4 Tab Servlet

With this Tab you can manupilate the presentation of your CTS URN when it's requested (e.g. by the function *GetPassage*)

- **deletexml**
  Documents can have XML tag elements on a word level that contain meta information. By enabling *deletexml* these elements will be deleted in the presentation of your document to make the texts better readable.

- **divs**
  Probably your document is structured (e.g. with paragraphe/sentence elements). If you select divs, the structure of you document and all the attribute of the structured elements (e.g. the n identifier) will be reproduced when requesting a Passage (GetPassage). When divs is not selected only the text will be reproduced. They may improve readability.

3 CTS Text Miner (CTSTM) and CTSTM Admin Tool

The CTSTM supports the analysis of textual data from CTS URNs with text mining techniques. Access the CTSTM Admin Tool via http://*[your VM's IP adress]*:8080/ctstm_admin/

3.1 Setting up a new CTS instance

click 'create new CTS' and specify a name

3.2 Tab DB-Config

Important Parameters:

- **db_user:** username of your MySQL Login
- **db_pw:** password of your MySQL Login

3.3 Tab Data Import

- **resetDataset**
  If set to true: all the datasets that were in de DB before are deleted.
  If set to false: you can add new datasets to the existing ones
- **cts**
  CTS URN source for your data
- **cts_configuration**
  see CTS configuration parameters on your Cheat Sheet or at http://ctstest.informatik.uni-leipzig.de/
- **docCount**
  number of documents you want to import from the source text collection
- **includeURNsWith** and **excludeURNsWith**
  Only URNs that include/do not include that substring will be added to your instance. The substring doesn't have to represent a complete element of the cts urn you want to import.
- you can leave the rest of the parameters as they are

3.4 Tab Servlet

Here you can set the default examples for the visualizations

3.5 Tab Browse Data

- in the Overview section you will find the URL for the Visualisation Tool to visualize your data

3.6 Create a new instance via SSH

Instead of using the CTSTM Admin Tool you can create a new instance by importing data into your VM via SSH commands

- copy a ctstm instance file (e.g. ctstm_demo) from your VM to your local drive:
  scp esu@*[server name]*:/home/esu/tomcat/webapps/ctstm_admin.war *[target directory]*
- access the war-file by unzipping it
- open the file *conf.properties* (ctstm_demo/WEB-INF/lib) with a text editor; set the important parameters (cf. 3.2, 3.3, 3.4); save and close
- open the file *settings.conf* (ctstm_demo/vis) with a text editor; change the URL according to your text collection; save and close

- rename your instance (former ctstm_demo); pack it into a ZIP-file and rename the file extension to *.war*.
- copy the WAR-file to your VM: scp *[source path of WAR-file]* esu@*[server name]*:/home/esu/tomcat/webapps/
- log into your VM
- navigate into the folder *lib*: cd /home/esu/tomcat/webapps/*[instance name]*/WEB-INF/lib
- to start the import of the documents into your instance execute the command: java -jar CTSTM.jar
- wait until import is completed
- you can now visualize your instance via the CTSTM Visualization Tool under the URL http://*[server name]*:8080/*[instance name]*/vis/