

TXM Reference Manual



version 0.5

Copyright © - ANR Textométrie - <http://textometrie.ens-lyon.fr/?lang=en>



This creation is distributed under a [BY-NC-SA Creative Commons license](https://creativecommons.org/licenses/by-nc-sa/4.0/)

Document Revision Table:

13/03/10	Serge Heiden (SH)	Creation
02/07/10	Matthieu Decorde	Update for release 0.4.7
15-29/07/10	SH	Rewrite for 0.4.7
27/08/10	SH	Section titles numbering, reorganized plan
08/10/10	Lauranne Bertrand	Update for release 0.5
19/01/11	SH	Corrections
11/03/11	SH	New section on import modules

TXM Reference Manual 0.5

Edition n° : 626
Content : 92 pp., 18578 occ., 78 ill., 9 tab.
Edition time: 03/11/11, 09:41:46 PM

Table of Contents

1	Preface	7
1.1	Who Should Use This Document	7
1.2	How This Document Is Organized	7
1.3	Related Readings	7
1.4	Accessing TXM Documentation On line	8
1.5	Typographic Conventions	8
2	Installing TXM	9
2.1	Requirements	9
2.2	Windows	9
2.3	Linux	11
2.3.1	Rapid installation	11
2.3.2	Classic installation	11
3	Getting to Know TXM	12
3.1	Starting TXM	12
3.1.1	On Windows	12
3.1.2	On Linux	13
3.2	Using Windows, Menus, Toolbars and Shortcut Keys	14
3.2.1	General Graphical User Interface	14
3.2.1.1	The explorer	15
	The Corpus view	15
	The File view and text editors	17
3.2.1.2	Commands	18
3.2.1.3	Icons	22
	Objects icons	22
	Commands icons	22
3.2.1.4	The Main Menus	23
	File Menu	23
	Corpus Menu	23
	Tools Menu	23
	Help Menu	24
3.2.1.5	The Results	24
3.2.1.6	The Messages	25
3.2.2	The Window Manager	25
3.3	Getting Help	26
3.4	Working with Corpora	26
3.4.1	Quick introduction	26
3.4.2	The complete story: Import, Export, Load corpora	26
3.4.3	Simple Import Commands	27
3.4.3.1	Raw Text Loaders	27
3.4.3.2	Raw XML Loaders	27
3.4.4	The Advanced Import Framework	28
3.4.5	Example of loader: the CNR+CSV Importer	29
3.4.6	Other Loaders	30
3.4.7	Saving & Exporting results	32

TXM Reference Manual 0.5

3.4.8	Sample corpora.....	32
3.4.8.1	DISCOURS corpus.....	32
3.4.8.2	QUETE corpus.....	33
4	Using TXM: commands.....	34
4.1	Describe corpus.....	34
4.2	Read Edition.....	35
4.2.1	Corpus.....	35
4.2.2	Partition.....	36
4.3	Build Sub-corpus.....	36
4.3.1	Simple sub-corpus building.....	36
4.3.2	Assisted sub-corpus building.....	38
4.3.3	Advanced sub-corpus building.....	39
4.4	Build Partition.....	39
4.4.1	Simple partition building.....	39
4.4.2	Partition building Assistant.....	40
4.4.3	Advanced partition building.....	42
4.5	Build Concordance.....	42
4.5.1	Queries.....	43
4.5.2	Browsing.....	46
4.5.3	Returning to text.....	46
4.5.4	Sorting.....	46
4.5.5	Word properties displayed.....	46
4.5.6	References displayed.....	46
4.5.7	Export.....	47
4.6	Cooccurrences.....	47
4.7	Lexicon and Index.....	49
4.7.1	Lexicon.....	49
4.7.2	Index.....	50
4.7.2.1	Properties combination.....	51
4.7.2.2	Queries.....	52
4.7.2.3	Thresholds.....	52
4.7.2.4	Browsing.....	52
4.7.2.5	Hypertext.....	53
4.8	Specificities.....	53
4.8.1	Partition specificities.....	53
4.8.1.1	Sorting.....	54
4.8.1.2	Graphics.....	55
4.8.1.3	Browsing the graphic.....	56
4.8.2	Sub-corpus specificities.....	56
4.9	Progression.....	57
4.10	Correspondence Analysis.....	58
4.11	Lexical table.....	60
4.12	TXM settings.....	63
4.13	Commands relationship.....	64
5	The Search Engine syntax.....	65
5.1	Quick introduction.....	65

6	Driving the TXM platform with scripts.....	68
6.1	Running Groovy scripts and commands.....	68
6.2	Running R scripts and commands.....	69
7	Import modules.....	70
7.1	Clipboard module.....	70
7.1.1	input.....	70
7.1.2	output.....	70
7.1.3	annotation.....	70
7.1.4	edition.....	70
7.2	XML-TEI BFM module.....	70
7.2.1	input.....	70
7.2.2	annotation.....	71
7.2.3	edition.....	71
7.3	XML-TXM module.....	71
7.3.1	input.....	71
7.3.2	output.....	72
7.3.3	annotation.....	72
7.3.4	edition.....	72
7.4	XML/w module.....	72
7.4.1	input.....	72
7.4.2	output.....	72
7.4.3	edition.....	73
7.5	Transcriber+CSV module.....	73
7.5.1	input.....	73
7.5.2	output.....	74
7.5.3	annotation.....	74
7.5.4	edition.....	74
7.6	Hyperbase module.....	74
7.6.1	input.....	74
7.6.2	annotation.....	75
7.6.3	edition.....	75
7.7	Alceste module.....	75
7.7.1	input.....	75
7.7.2	output.....	75
7.7.3	annotation.....	75
7.7.4	edition.....	75
7.8	CNR+CSV module.....	75
7.8.1	input.....	75
7.8.2	output.....	76
7.8.3	annotation.....	76
7.8.4	edition.....	76
7.9	TXT+CSV module.....	76
7.9.1	input.....	76
7.9.2	output.....	77
7.9.3	annotation.....	77
7.9.4	edition.....	77

TXM Reference Manual 0.5

8 Keyboard Shortcuts.....	78
8.1 Text Editor.....	78
8.2 Graphics Output.....	80
8.3 Windows.....	81
9 TXM Glossary.....	82
10 Bibliography.....	88
11 Index.....	89

1 Preface

1.1 Who Should Use This Document

If you want to use the TXM platform, this document will introduce you step by step to the different concepts of the software and to the different tools available to analyze various textual corpora.

If you want to adapt the TXM platform to specific corpora, this document will also introduce you to the scripting environment available to customize the import system.

1.2 How This Document Is Organized

This document first describes how to install the software on various platforms and how to start it.

Then it describes how the user interface is organized and how to import a new corpus into the platform.

The next section describes the available tools and how to use them to analyze a corpus.

The way to extend the platform with the scripting environment is then introduced.

[The document ends with reference appendix : a glossary of notions and an index. TO BE DONE]

1.3 Related Readings

The official Textométrie project web site publishes all the documentation related to the TXM platform : <http://textometrie.ens-lyon.fr/spip.php?article98&lang=en> (screencast tutorials, textometry methodology fundamental documents, textual encoding related documents, search engine and statistical engine related documents and reference documents for the scripting engine).

It is also the reference site for all scientific publications related to the project : <http://textometrie.ens-lyon.fr/spip.php?article82&lang=en>

The TXM Wikis are the best place to share knowledge about the platform usage with other users and with developers :

- [EN] The international English language wiki is at <http://textometrie.sourceforge.net>

(please subscribe to Sourceforge¹ and ask for permission to be able to edit the wiki)

¹ <http://sourceforge.net/account/registration>

- [FR] The French language wiki is at <https://listes.cru.fr/wiki/txm-users/en/startup>

(please subscribe to the 'txm-users'² mailing list to be able to edit the wiki)

The French language wiki currently has the following structure:

- bug reports on the RCP version : from mails and meetings ;
- bug reports on the GWT version
- new features asked for the RCP version
- that wiki also allows you to participate to the writing of the documentation or to translations.

If you want to add or modify core functionalities to the software, that is to change the sources of the software, you should also read the TXM Developers Guides referenced by the developer wiki (<https://sourceforge.net/apps/mediawiki/textometrie>), with the Javadoc and the R module documentation.

1.4 Accessing TXM Documentation On line

This document, and its translations, are always available at the address :

<http://sourceforge.net/projects/textometrie/files/documentation>

1.5 Typographic Conventions

In that documentation, some specific items are distinguished by a different typography:

- sample literal strings are rendered in `Courier` : directory paths, file names, sample queries , strings or links
- **Arial** rendering is reserved to section titles
- Arial rendering commands

² <https://listes.cru.fr/sympa/subscribe/txm-users>

2 Installing TXM

2.1 Requirements

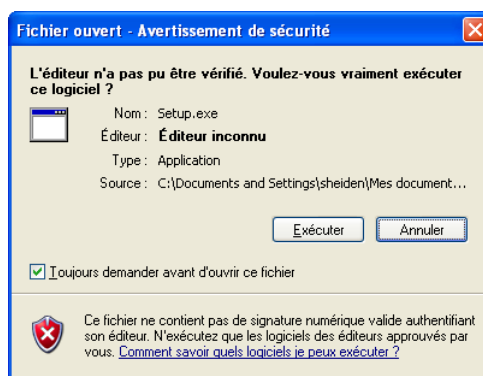
This version of the software is compatible with Windows and Linux³.

The following resources are recommended :

- 170 Mb of disk space for installation;
- 350 Mb of memory for execution.

2.2 Windows

1. First, download the file “txm_0.5_win.exe” at the address : <https://sourceforge.net/projects/textometrie/files/software/0.5>
2. Execute the file by double-clicking on it :
 - a. Depending on the security level of your Windows operating system, the following dialog box may pop up (in your language):

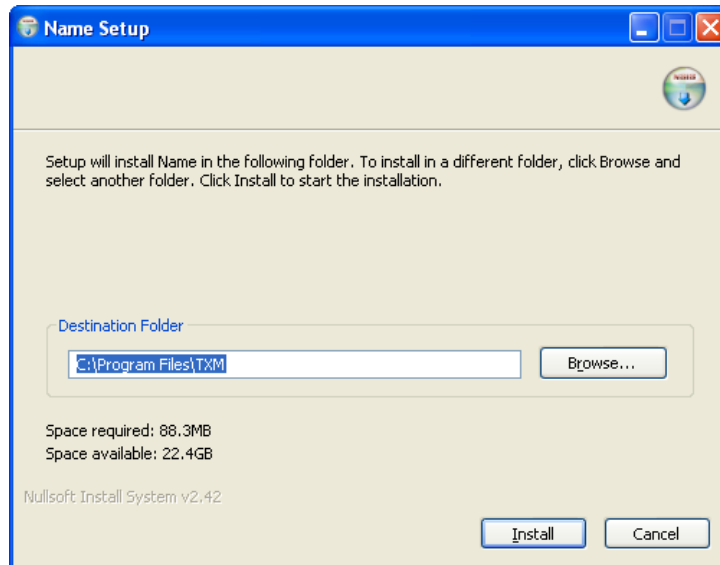


In that case, please click on the left button “Exécuter” (Execute)

³ Only Windows XP, Vista & Seven, and Linux Ubuntu have been tested for this release.

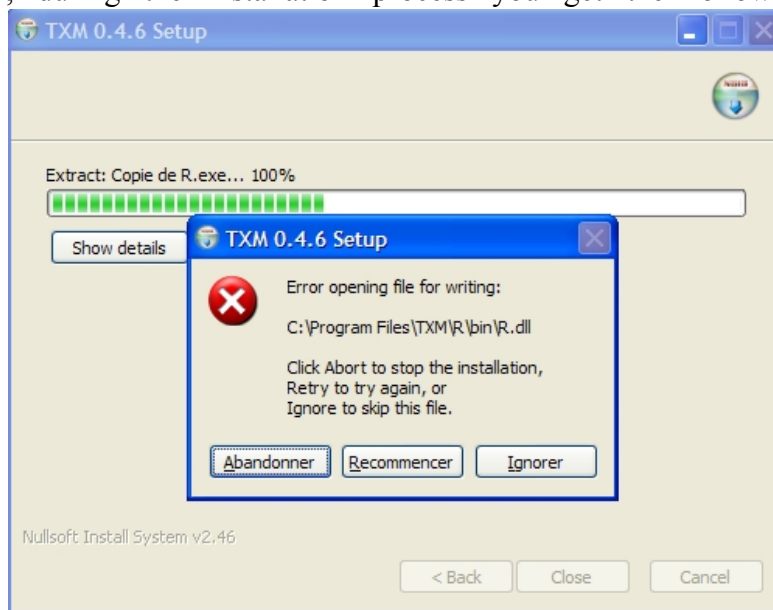
TXM Reference Manual 0.5

- b. In the next dialog box:



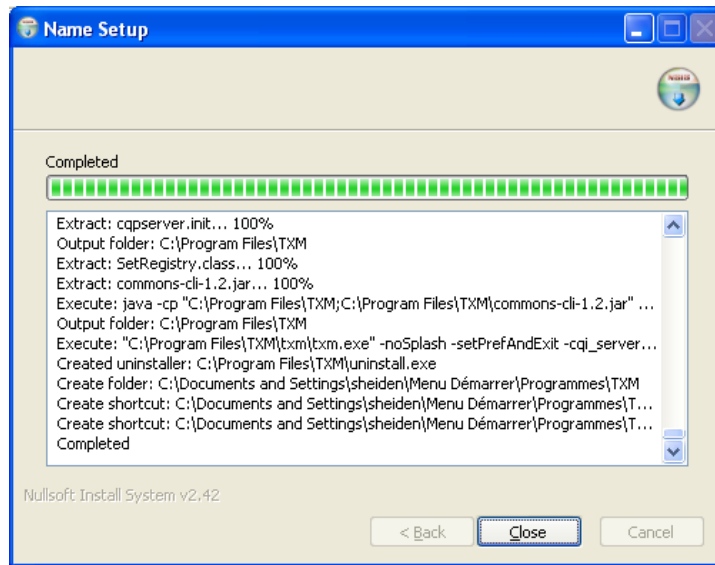
Click on “Install” (you may choose another install directory before).

- c. The install process takes about a minute.
d. If, during the installation process you get the following message :



This means that an Rserve process is still running on the machine and that the install process cannot modify its binary file. You must then 1) quit TXM or kill the Rserve.exe process running from the Process Explorer, and 2) click on 'Recommencer' (Restart) to resume the install process.

- e. In the next dialog box:



Click on “Close”

- f. Installation is now completed.

2.3 Linux

2.3.1 Rapid installation

1. Download the file “txm_0.5.deb” at the address:
<https://sourceforge.net/projects/textometrie/files/software/0.5>
2. Launch the “txm_0.5.deb” file to start the installation process with the gdebi package manager
3. Launch TXM through the “Applications / Sciences / TXM” menu item of your system menu

2.3.2 Classic installation

4. Download the file “txm_0.5_linux.tar.gz” at the address:
<https://sourceforge.net/projects/textometrie/files/software/0.5>
5. Click on “txm_0.5beta_linux.tar.gz”
6. Extract the content of the archive in a directory (you can use the command line "tar xvf txm_0.5beta_linux.tar.gz")
7. Go to that directory
8. Run : “bash install.sh <path to the directory where TXM is installed>”
(the INSTALL file contains more informations on the Linux install process)
Run TXM with the command : “TXM&” or with the ALT+F2 shortcut followed by “TXM”

3 Getting to Know TXM

The current TXM platform prototype helps you to build and analyze tagged and structured corpora :

- it helps you to import your textual resources to build a corpus from various format, or directly from any text copied in the clipboard.
- it builds subcorpora from various specifications of textual units properties
- it builds partitions from specification of properties
- it builds an HTML edition for each textual unit of a corpus
- it computes the whole vocabulary of a corpus or lists various combinations of word property values
- it builds lexical tables from partitions or index
- it searches complex lexical patterns based on lexical units properties and builds kwic concordances of the matches. From any line in a concordance, you can get to the edition page containing the corresponding keyword
- it computes cooccurrents around complex lexical pattern
- it computes the specificity model of occurring words or tags inside a partition or a sub-corpus
- it computes the factorial correspondence analysis of word properties inside a partition.

The software is composed of four components :

- a full text search engine;
- a statistics engine;
- an import environment;
- a scripting engine.

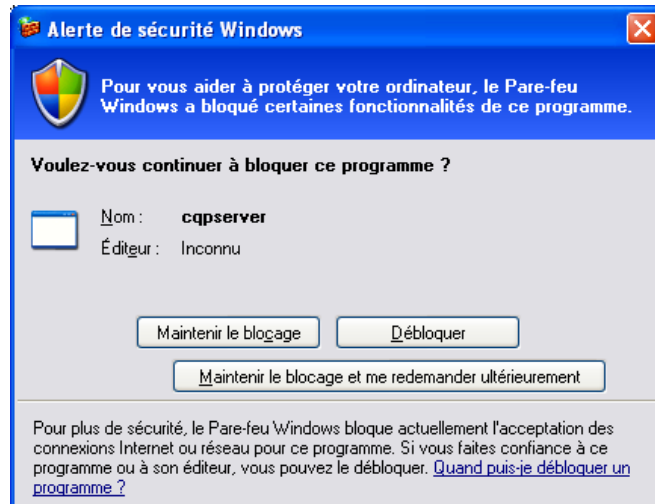
This manual will introduce you to each component through the various commands available in the platform.

3.1 Starting TXM

3.1.1 On Windows

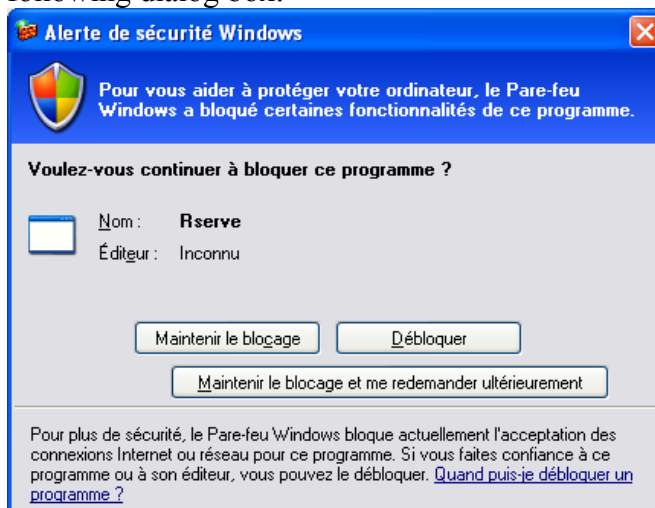
1. In the menu “Start” / “All Programs” / “TXM” select “TXM”
2. For the first start, depending on the level of security of your Windows operating system, you may have to answer some security alerts in the following way:

- a. In the following dialog box:



Click on “Unblock / Débloquent”⁴

- b. In the following dialog box:



Click on “Unblock / Débloquent”⁵

3.1.2 On Linux

1. Through the “Applications / Sciences / TXM” menu item of your system menu
2. Or call in a shell : TXM&
3. Or with the ALT+F2 shortcut followed by “TXM”

⁴ The 'cqpserver' process is the textual database engine which needs to communicate with the TXM platform through a network protocol.

⁵The 'Rserve' process is the statistics engine which needs to communicate with the TXM platform through a network protocol.

3.2 Using Windows, Menus, Toolbars and Shortcut Keys

3.2.1 General Graphical User Interface

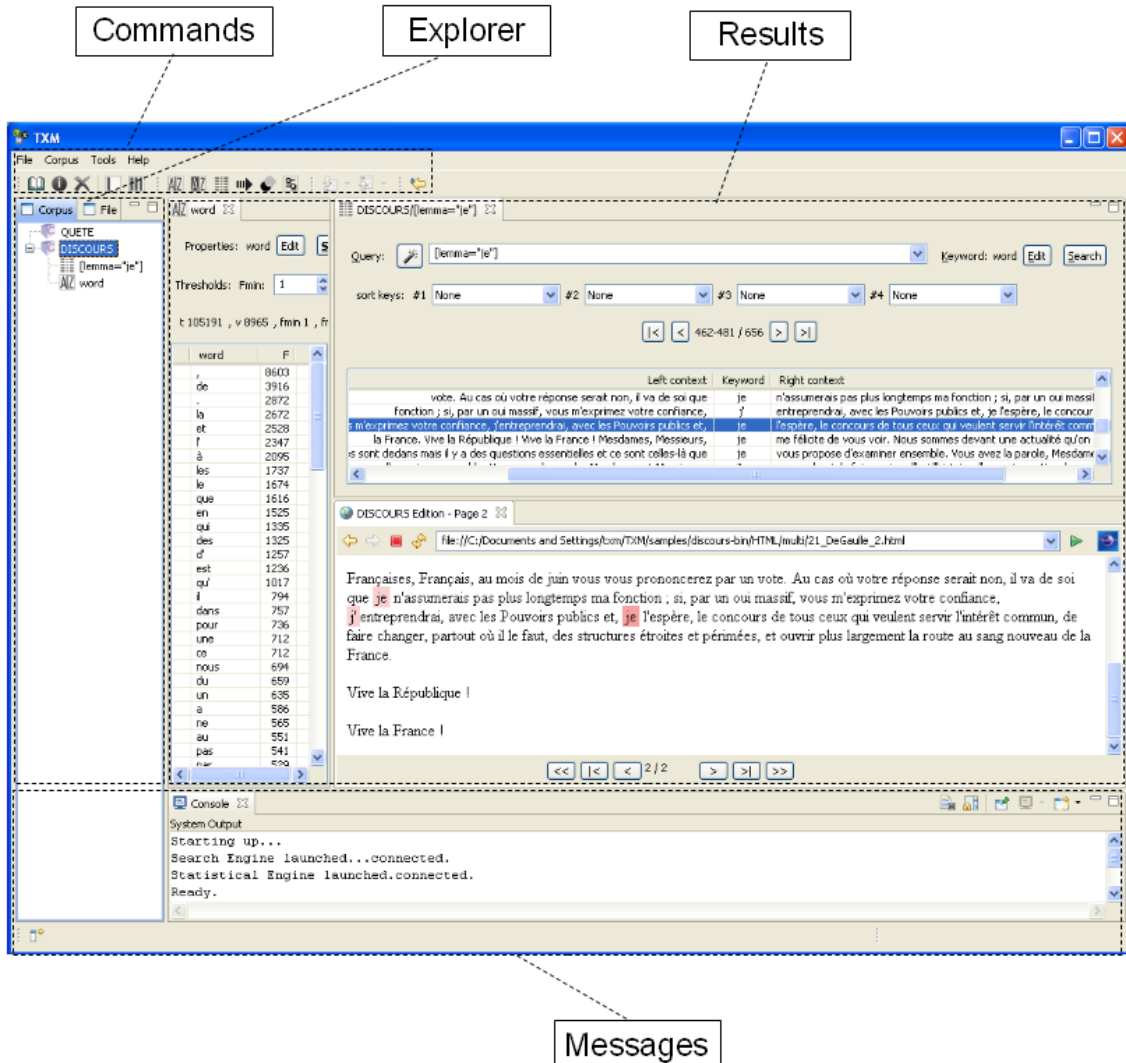


Illustration 1: The general interface of TXM

The user interface of TXM is divided in four main zones depicted in illustration 1 :

- the explorer : root corpora, results of commands, scripts icons. In fact, all objects which are managed by TXM and produced by commands ;
- the commands : where actions on objects are expressed;
- the results : the output windows;
- the messages : the comments from commands execution.

All the zones are managed by a single window manager.

We will first present the main zones and then present how to organize the interface with the window manager.

3.2.1.1 The explorer

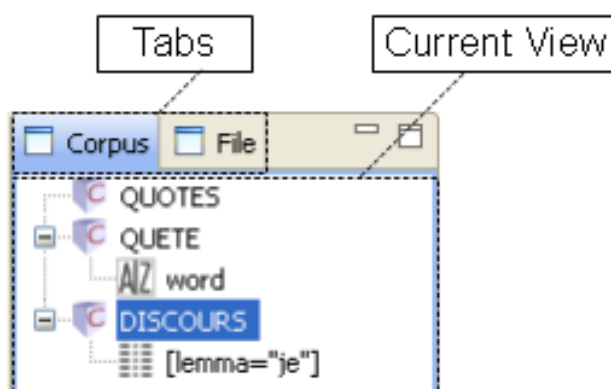


Illustration 2: The explorer.

The explorer is the main place for the user to select the objects on which to apply the commands of TXM and to get to the results of the commands.

The explorer is organized in two different views :

- the "Corpus" view : related to available corpora for analysis;
- the "File" view : related to files found on the file system to edit.

Each view is accessed by its specific tab.

The Corpus view

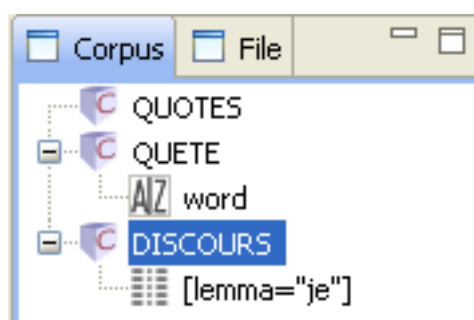


Illustration 3: The Corpus view.

The Corpus view displays all the different corpora available for analysis within TXM and all the icons of the objects built by TXM during a work session. The corpora have been created by the Import command from the File menu.

The Corpus view is organized hierarchically. Each root object is an independent Corpus. That corpus is related to the Base from which the texts were imported. All the children icons are objects resulting from TXM commands :

- Subcorpora (“C” icon, same as the 'root' corpus) from 'Create sub-corpus';
- Partitions (“P” icon) from 'Create partition';
- Lexicon;
- Index;
- Concordance;
- Cooccurrences;
- Specificities ;
- Correspondence analysis;
- Lexical table.

A branch in the tree results from new objects being created as results of commands applied to the parent object.

Each object type can be applied on a specific logical set of commands :

- a “Corpus” object can be applied on any command ;
- a “Sub-Corpus” object can be applied on the same commands as the corpus, plus the Specificities command.
- a “Partition” object can be applied on only a Specificities, Factorial analysis or Lexical table command.

Double-clicking on result objects reopens the results window when it has been closed.

The File view and text editors

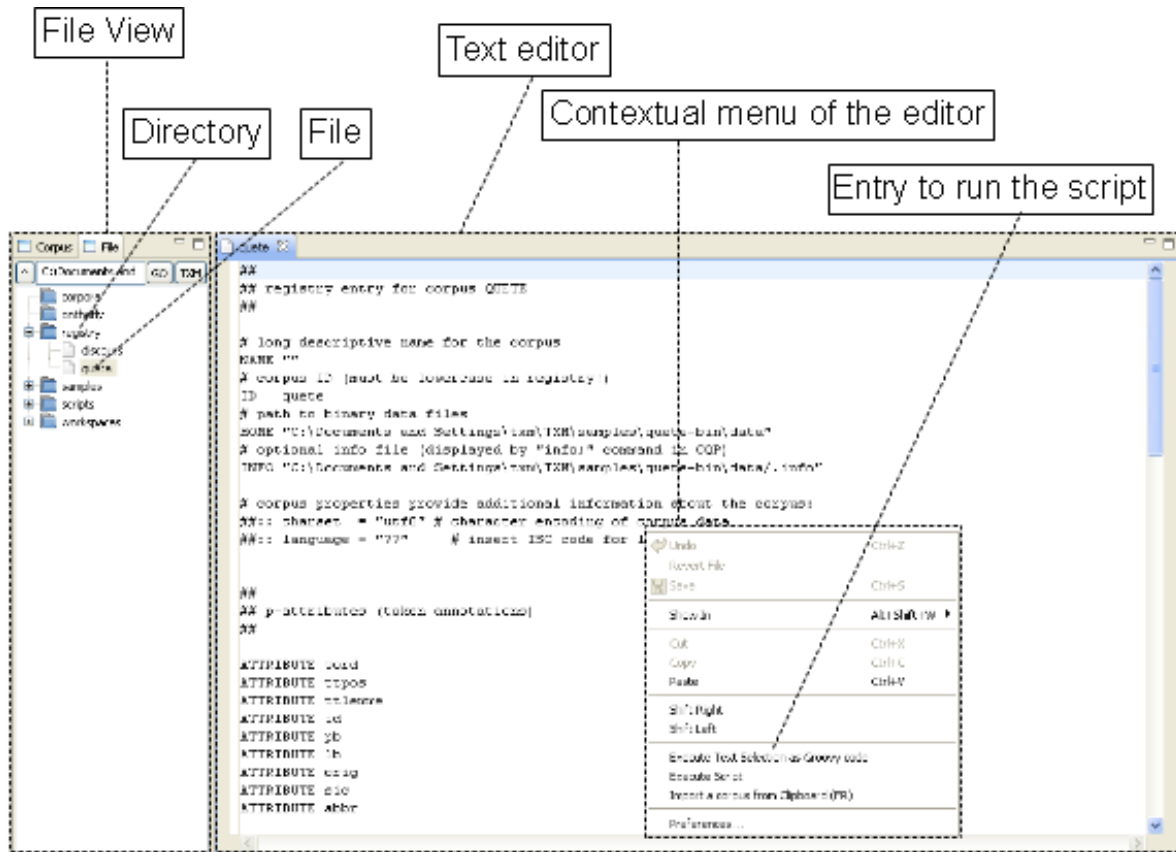


Illustration 4: The File view.

The File view displays a classical hierarchical icon view of the folders and files in the file system⁶. It allows you to edit all those files from inside TXM (TXT or XML source files, Groovy or R script files, etc.), should you need to correct an input file or a script for example.

Browsing

The “^” button opens the parent directory of the current directory.

The text field display the current directory, you can change it and press “Enter” or click on the “OK” button to refresh the view.

The “TXM” button brings back to the TXM user's directory.

A double-click on a directory expands its content.

A double-click on a file icon opens it in a new text editor window. The same result is obtained through the 'Open File' command in the 'File' menu.

Editing a text

⁶ The default path of that view is the user's TXM home directory (that is \$HOME/TXM).

In a text editor, the text can be modified, saved, etc. by usual commands : select/copy/paste, search&replace, save, etc.

Please see the section 6, 'Text Editor Shortcuts', for the list of available editing commands.

If the text is a Groovy script, it can be executed directly with the 'Execute a Groovy script' command in the context menu (right click on the text). You can also execute only a selection of the text with the 'Execute the selection as a Groovy script' command in the context menu. See the 5th section 'Scripting the TXM platform' for more information on the scripting environment embedded in TXM.

If the text is an R script, it can be executed directly with the 'Execute an R script' command in the context menu. You can also execute only a selection of the text with the 'Execute the selection as an R script' command in the context menu.

3.2.1.2 Commands

In TXM, main commands are expressed through three different but equivalent ways :

- 1) when an object icon is selected in the objects zone, the user can execute a command on that object by clicking on the corresponding command button in the **Toolbar**

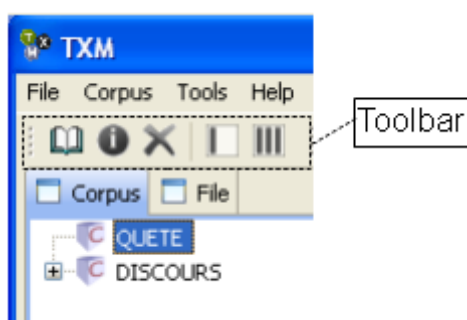


Illustration 5: The Toolbar.

- 2) when an object icon is selected in the explorer, the user can execute a command on that object by selecting the corresponding command in the upper "File", "Corpus" or "**Tools**" Menus

- a. The “File” menu and its Export command :

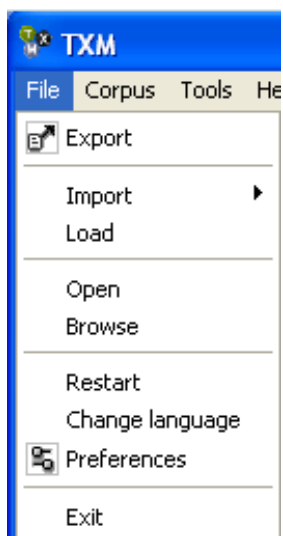


Illustration 6: The File menu

- b. **The “Corpus” menu** and its description and corpus' manipulations commands :

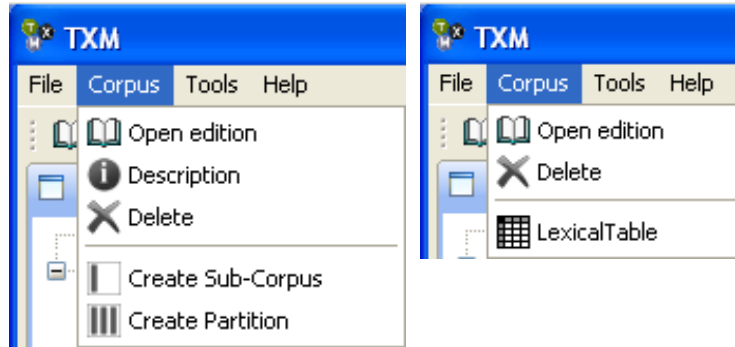


Illustration 7: The “Corpus” menu with, on the left, the corpus commands and, on the right, the partitions commands.

The menu configuration changes with the type of the icon selected : for the first menu a corpus is selected, for the second one, it is a partition.

- c. **The “Tools” menu** gives access to the textometric tools :

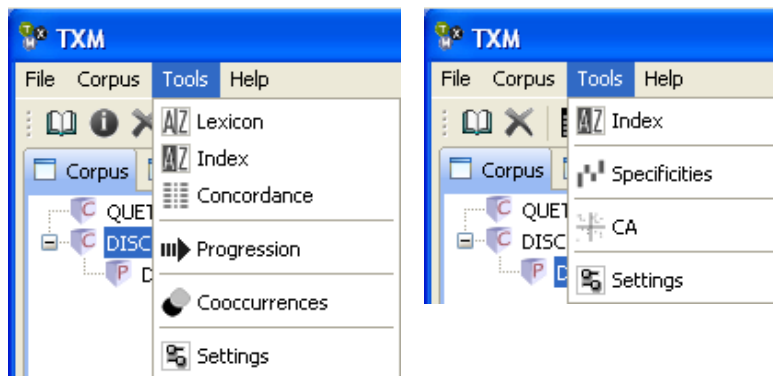


Illustration 8: The “Tools” menu, for the corpus and the partition objects

- 3) the user can open the **Contextual Menu** by clicking on the right button of the mouse on the object to apply the command on

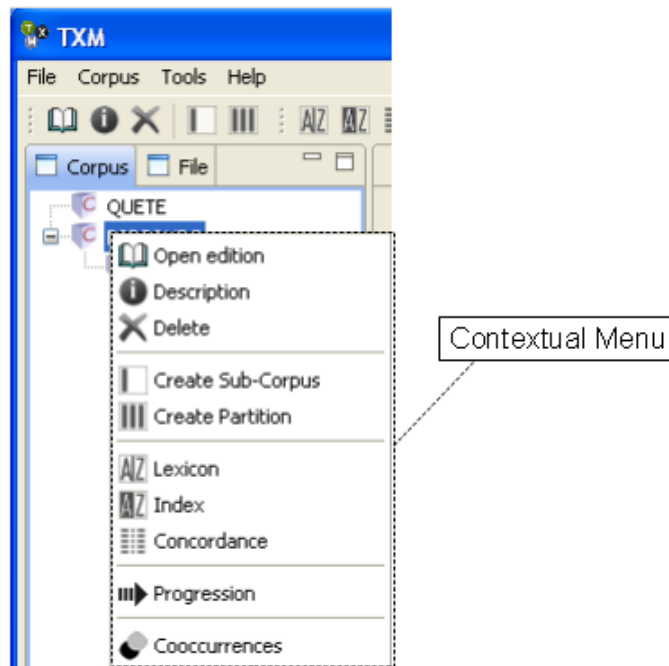







Illustration 9: The Corpus Contextual Menu.

The commands are described in detail in the section 4 'Using TXM: Commands'. All results windows can also give access to commands depending on the object types contained in the result.















3.2.1.3 Icons

Here is the list of all the icons used in the TXM graphical interface :

Objects icons

	Corpus
	Partition
	Open edition
	Progression
	Lexical table

Commands icons

	CA
	Concordance
	Cooccurrences
	Create Partition
	Create Subcorpus
	Delete
	Description
	Export
	Index
	Lexicon
	Query assistant
	Search
	Settings
	Specificities

3.2.1.4 The Main Menus

Here is the description of all the available main menus in TXM in the upper left part of the interface:

File Menu

- Export : exports a result at least as raw text
- Import: imports a new corpus from its sources with one of the available import loaders
 3. From clipboard : imports the text from the clipboard
 4. ...
- Load: loads a new corpus from its binaries directory
- Open: opens a file in a new text editor
- Browse: opens a file in the integrated web browser
- Restart: restarts TXM search and statistics engines
- Change language : shows a window to changing the interface language of TXM as set in the preferences menu: Preferences > TXM > User > Language
- Preferences : To set various parameters of TXM, like some threshold calculation (minimal frequency, etc.).
- Exit: quit the application

Corpus Menu

- Open edition: displays the first page of the edition
- Description: displays the structures and the word properties available
- Delete : deletes the selected object
- Create sub-corpus: builds a new sub-corpus
- Create partition: builds a new partition
- Lexical table : creates a lexical table from a partition or a partition index

Tools Menu

- Lexicon: lists all the different values of a specific property of words with their overall frequency
- Index: lists the different values of combination of different word properties with their overall frequency from the results of a specific CQP query
- Concordance: searches for patterns of a CQP query expression and display results as kwic concordances
- Progression : displays evolution of one or more patterns throughout a corpus
- Cooccurrences : computes cooccurrences from a CQP query
- Specificities: lists the positive and negative specificity scores of a specific property of words for each part of a partition
- Correspondence Analysis: draws texts and word properties on the factorial map of the first two factors obtained by factorial correspondence analysis on a partition
- Settings : opens parameters page of TXM tools. [In this version, this is the same as the 'File Settings' menu].

Help Menu

- Key Assist : displays all the available keyboard shortcuts
- Report a bug : opens the “report a bug” web page
- Ask for enhancement : opens the “ask for feature” web page
- Submit to txm-users list : opens the “submit” page of the 'txm-users' mailing list
- Check for update : opens the Sourceforge TXM download page
- Install TreeTagger : opens the TreeTagger install tutorial page
- About : displays TXM version number and license informations

3.2.1.5 The Results

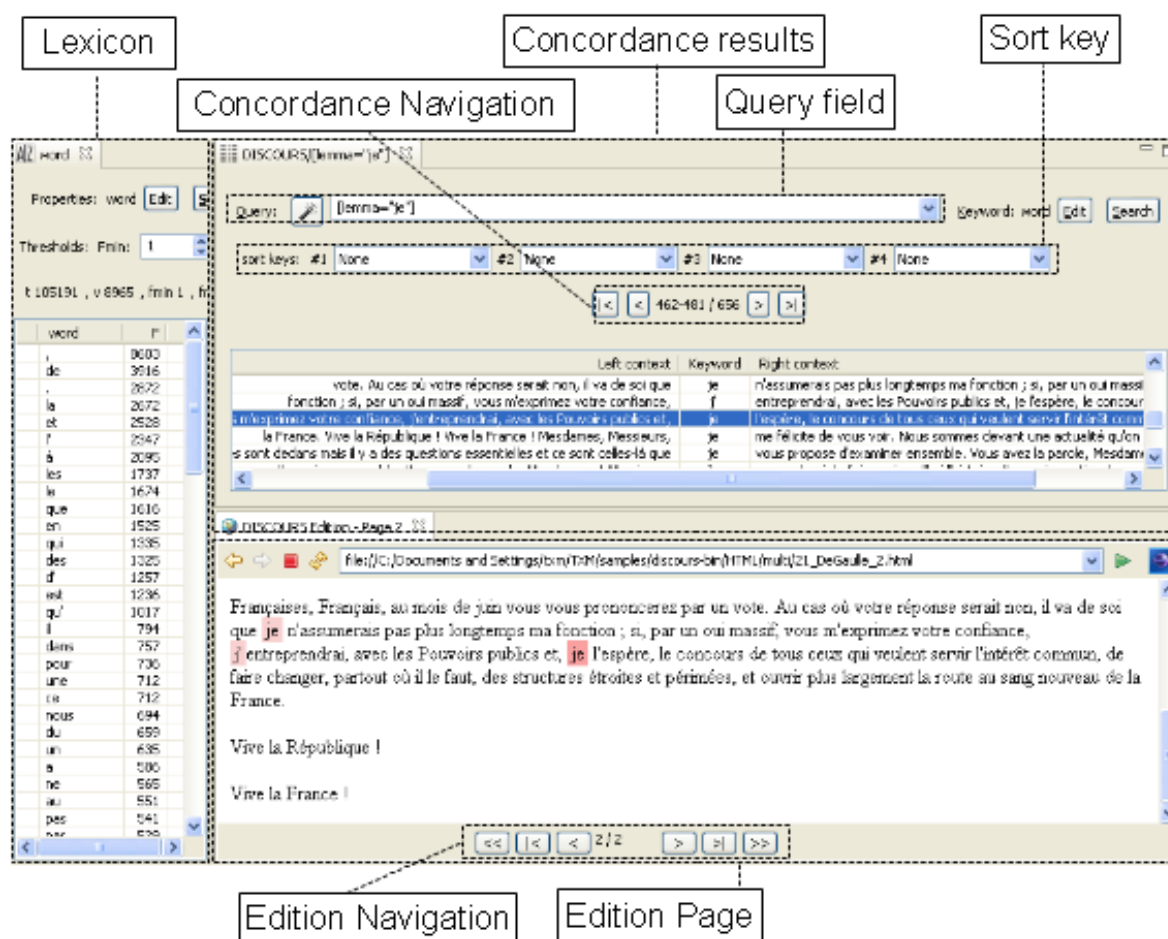


Illustration 10: The results.

All the results of the commands are by default displayed in the right results zone. First each command, the results are displayed in a new window, with a name related to the command and its parameters, and a new icon is added in the corpus view.

The name of the window is displayed in the tab of the window and in the legend of the icon. That tab is an important control widget to manage the display of the window as will be seen in the window manager section.

If a window is closed during the session, it can be reopened by double-clicking on the corresponding icon in the corpus view.

3.2.1.6 The Messages

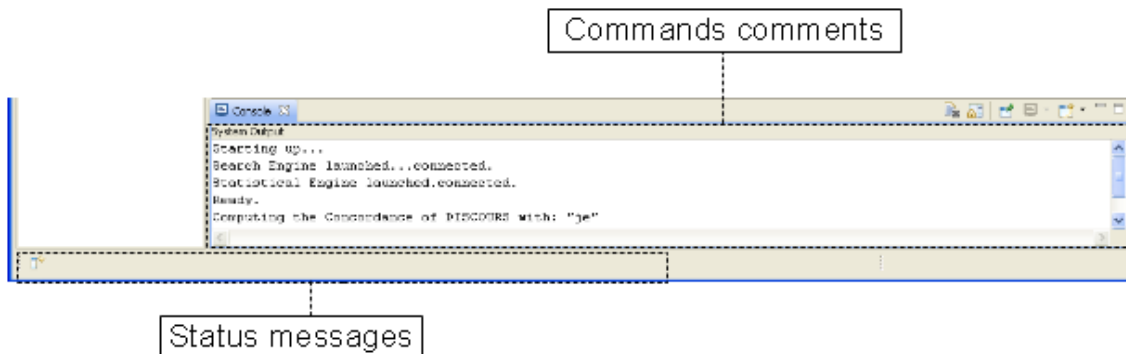


Illustration 11: The Messages zone.

The Status line display simple messages, like the number of results.

The commands comments area displays more informations related to commands, it can be scrolled, selected, copied and pasted. It can also display critical messages.

3.2.2 The Window Manager

With the window manager, one can maximize, minimize, collapse, reopen, move and resize any window of the interface with the mouse efficiently.

The window manipulations are the following :

- temporarily maximize the window to full screen: double-click on the window tab;
- put the window back to its original size: double-click on the window tab;
- move and resize the window depending on the place it is dropped: drag the tab of the window to the place it should go. Before releasing the mouse, when it arrives at the center of the outer limit of the underneath window, called a hot spot, a ghost window frame is drawn to show the size and the place the window will have if the user releases the mouse there. Each middle border of the underneath window has a hot spot to choose:
 - left: to split vertically and let the window on the left side;
 - right: to split vertically and let the window on the right side;
 - up: to split horizontally and let the window on the top side;
 - bottom: to split horizontally and let the window on the bottom side.
- minimize the window: click on the "Minimize" icon of the window;

Each interface zone "objects" and "results" manage its windows in a coherent way.

The current window layout is always saved automatically by TXM.

3.3 Getting Help

[The text of that manual will be embedded into the TXM platform as an implicit corpus with its own edition. TO BE DONE]

3.4 Working with Corpora

3.4.1 Quick introduction

With TXM you can analyze textual data coming from various sources :

- the “File / Import / From Clipboard” command allows you to use the TXM platform commands on any text you have selected then copied from common desktop applications : Firefox, Thunderbird, Writer, etc.
- the “File / Import / Directory” command allows you to analyze a set of raw texts found in a single directory;
- the “File / Import / XML Structure” command allows you to analyze a set of XML encoded texts found in a single directory;
- other entries from the “File / Import” menu allows you to analyze corpora in various specialized formats like the Hyperbase format or the XML TEI P5 format.

The platform is released with two ready to use sample corpora⁷:

- DISCOURS: a set of French Presidents speeches transcriptions;
- QUETE: an edition of the Ms K manuscript of the “Holy Grail” text written in the old French language.

The next section presents in detail all the available commands to import corpora in the TXM platform.

3.4.2 The complete story: Import, Export, Load corpora

The TXM platform can work on corpora of various formats : from simple raw text files to densely XML TEI P5 encoded ones.

- **Import** : To be able to work on a specific corpus, it has to be imported into the TXM platform with one of the commands of the “**File / Import**” menu. Each command analyzes specific corpora sources to build all the necessary elements for TXM to work on it. It can take some time depending on the size of the corpus and the complexity of the loader. When that process has been done, the corpus will be available instantaneously for all the next working sessions with TXM until the corpus is deleted (you don't have to import the corpus again for each working session). The corpus is added to the “Corpus” view. The next section will introduce you to all the available loaders in this release.

⁷ Please read the “sample corpora” section for their full description.

- **Export** : To transmit a corpus already imported into the platform to another TXM installation (say on another machine), you can copy the directory corresponding to the binary corpus built by TXM. That directory is located at “\$HOME/TXM/corpora/<name of the corpus>”. As a byproduct of the import process, several intermediary source files encoded in the “TEI-TXM”⁸ format have been produced in the “\$HOME/TXM/corpora/<name of the corpus>/txm” directory. Those files can be used as an XML interchange format with other tools.
- **Load** : If you have copied a corpus directory from another TXM installation, you can load it directly into TXM with the “**File / Load**” command. That command is faster than the Import command. You only need to call it once for a TXM installation.

3.4.3 Simple Import Commands

3.4.3.1 Raw Text Loaders

The “From Clipboard” and “Directory” entries of the “File / Import” menu import simple raw texts, without any XML tags in them. Each word is tokenized and annotated with a part-of-speech property and a lemma property⁹:

- **From clipboard loader** usage :
 1. Select then copy some text from an application (OpenOffice Writer, Thunderbird, Firefox; etc.)
 2. Select the command “File / Import / From Clipboard”
 3. A corpus named “ClipboardN” is added to the corpus view, where N is the current clipboard import number in the current session.
- **Directory loader** usage :
 1. Select the command “File / Import / Directory”
 2. In the popup form, select the directory containing the raw text source files. Note : each source file will be imported as a textual unit, it must have the extension “.TXT” to be considered by the import process. This command imports all the files contained in the selected folder tree (folders and sub-folders).
 3. A corpus with the same name as the directory will be created in the corpus view.

3.4.3.2 Raw XML Loaders

The “XML structure” entry of the “File / Import” menu imports a single valid XML file into TXM. Each tag will be interpreted as a structural unit with properties coming from the tag attributes. Each word is tokenized and annotated with a part-of-speech property and a lemma property. The `text` tag is not interpreted by the tokenizer, so be sure to remove it before the import process¹⁰ :

- **XML structure loader** usage :
 1. Select the command “File / Import / XML Structure”

⁸ The TEI-TXM format is an extension of the XML TEI P5 format. Its schema is not publicly released yet.

⁹ By default, TreeTagger is used to tag words with the French model, but you have to install it yourself because of TreeTagger licensing conditions (see the tutorial displayed by TXM if it is not installed yet).

¹⁰ This “bug” will be removed in a next release.

2. In the popup form, select the directory containing the raw XML source files.
Note : each source file will be imported as a textual unit, it must have the extension “.XML” to be considered by the import process.
3. A corpus with the same name as the directory will be created in the corpus view.

3.4.4 The Advanced Import Framework

The TXM platform is designed to import various kind of source corpora.

To ease the design of specific corpus loaders, several key concepts have been defined to specify the import process¹¹ :

- a **document unit** represents a body of textual data for which all the metadata are the same;
- the **metadata** of a document unit is a simple set of properties having simple values (title, date, author's name, domain, type...);
- a document unit is organized as a tree of **structural units**;
- each node can have any number of **properties** having simple values;
- the leaf nodes of a document unit are the **lexical units** (words);
- an **NLP tool** can be applied to any source file during the import process (like a tagger);
- each document unit can have one or several **editions** built.

An import process, or loader, creates these key concepts into the TXM platform, from the informations found in corpus sources.

Those building elements can be :

- in a single file or in several;
- in different formats.

The import process of a corpus - from the sources into the search engine indexes, editions, etc. - is implemented by a Groovy script.

Any Groovy script, as any import loader, can be plugged into the TXM platform at run time.

The input parameter of a loader is the root directory of the source corpus.

The output of a loader is loader dependent but at least a new root object for the corpus is added to the “Corpus view” to be able to apply any TXM command on it, and a new directory is created to hold the binary version of the corpus at “\$HOME/TXM/corpora/<name of the corpus>”.

¹¹ See the “Import Environment 0.4.7 (FR)” document for an introduction to all the available concepts.

You can see below the import setup window :

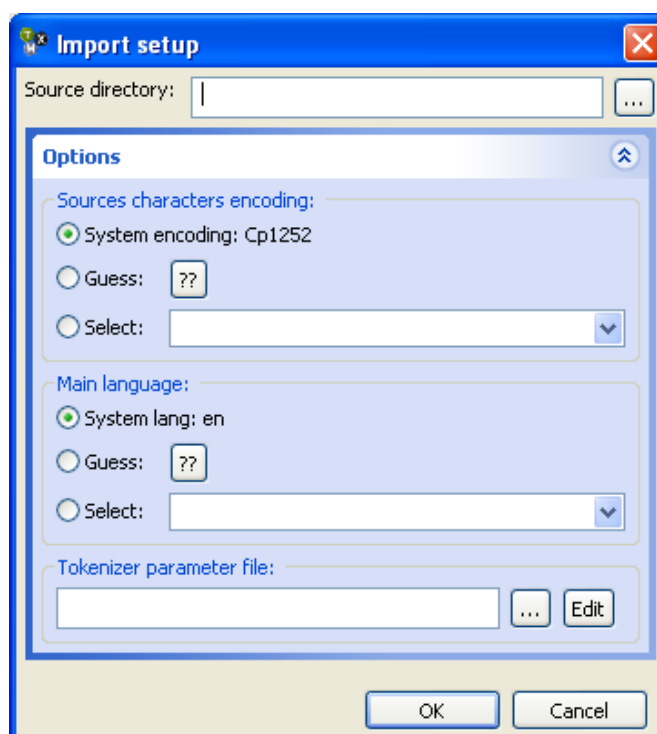


Illustration 12: Import window.

The “source directory” parameter is mandatory.

You can check “system encoding” (default), check “guess”¹² (and press the “??” button) or select directly the encoding of your source. You can do the same for the main language.

3.4.5 Example of loader: the CNR+CSV Importer

The CNR+CSV reads a source corpus in the following format :

- each document unit is in a single file;
- the format of the document unit file is “CNR” : that format is the output format of the commercial software “Cordial” which is a French tagger and lemmatizer. That format is like the CSV format (column separator being the tabulation character);
- all the metadata are stored in a single Excel table (the import process uses a CSV export format of that table). Each metadata is defined in one column. All the metadata of a single document unit are on the same line;
- the only structural unit recognized and encoded is the sentence level, which comes from the Cordial tagger output;
- lexical units have the properties encoded in the CNR output columns (word form 'word', lemma 'lem', and part of speech 'pos').

That loader can be applied to the sources of the sample DISCOURS corpus found in the distribution.

¹² This function is not available in this version.

The results of the loader are :

- a new root corpus is added to the “Corpus” view giving access to any TXM commands on it;
- two different HTML editions are produced for each text : one paginated every 200 words and one in a single file. In those editions, each word has a flyover displaying its properties;
- search engine indexes have been compiled.

In the next section, you will find a synthetic description of the loaders and the recommended information to write in the dialog box.

3.4.6 Other Loaders

The TXM platform can already import several other corpus formats through different loaders:


Name	Document Unit	Main Format	Metadata	Structural Units	Lexical Properties	Editions	Recommended information
CNR+CSV	Single text per file	Cordial Multitext	In “metadata.csv” file	s	word, pos, func, lemma (FR)	Single, paginated every 200 words	System encoding
Hyperbase	Several texts (in a single file)	Hyperbase (old format)	None	s	word, pos, lemma (FR)	Single, paginated every 200 words	System encoding
Alceste	Many texts (in a single file)	Alceste	Analytic	s	word, pos, lemma (FR)	Single, paginated every 200 words	System encoding
Transcriber+CSV	Single transcription per file	Transcriber (XML-TRS)	In “metadata.csv” file ¹³	Sections, speech turn-taking	word, pos, lemma, spk, event	Paginated every 200 words after a speech turn	
XML-TEI BFM	Single text per file	XML TEI P5	Bibliographic	s and some other	word, pos (AFR)	paginated on <pb/>	

¹³ Metadata is associated to one transcription and to only one of its speakers.

TXM Reference Manual 0.5

Name	Docum ent Unit	Main Format	Metadat a	Struct ural Units	Lexical Properti es	Editions	Recommen ded information
				BFM units			
XML-TXM	Single text per file	XML TXM	None (should already be encoded inside the source)	Any XML tags	word, pos, lemma	Single, paginated every 200 words	
TXT+CSV	Single text per file	TXT : raw text	In “metadata .csv” file	None	word, pos, lemma	Single, paginated every 200 words	System encoding
XML/w	Single text per file	XML	None (should already be encoded inside the source)	Any XML tags	word, pos, lemma	Single, paginated every 200 words	

3.4.7 Saving & Exporting results

Each result of a TXM command (lists, tables, graphics) can be exported in a file. That file is at least in the CSV format for tables and in the SVG format for graphics. The export command can be accessed in the contextual menu of the result icon in the “Corpus” view or through the “Export”  button in the toolbar when the result object is selected.

3.4.8 Sample corpora

The TXM platform is released with several sample corpora encoded in representative formats that the platform can process. They are released under a BY-NC-SA Creative Commons license.

3.4.8.1 DISCOURS corpus

The “DISCOURS” corpus has been released by Damon Mayaffre from the BCL (CNRS) laboratory in Nice, France. It is composed of 29 discourses produced by:

- two French presidents: Pompidou (5 discourses) and de Gaulle (24);
- between 1958 and 1971;
- of types: “Allocution radiotélévisée” – speech on tv (14), “Entretien radiotélévisé” – speech on radio (3), “Conférence de presse” – press interview (11)

Each discourse has been tagged with the Cordial tagger with the usual Hyperbase software parameters. The tagset is the Multext tagset (described in the Weblex manual at <http://weblex.ens-lsh.fr/doc/weblex/cordialtagset.html>).

The importation of the corpus into the TXM platform encoded the following objects:

- structural units: `discours / s` (for sentence)
 - each “discours” unit has the following properties encoded:
 - `date`
 - `loc`: the name of the president
 - `type`
 - each lexical unit as the following properties:
 - `word`: the graphical form;
 - `pos`: the Cordial part of speech tag;
 - `lem`: the Cordial lemma;
 - `func`: the Cordial syntactic function code;
 - `sent`: the sentence number.

3.4.8.2 QUETE corpus

The “QUETE” corpus has been released by Christiane Marchello-Nizia and Alexei Lavrentiev. It is based on their critical edition of the “Queste del saint Graal” from the Ms K manuscript “Lyon, Bibliothèque municipale, Palais des arts 77”.

In that text, each word is tagged by a morphosyntactic tag of the CATTEX2009 tagset for old French (http://bfm.ens-lyon.fr/article.php3?id_article=176).

The importation of the corpus into the TXM platform encoded the following objects:

- structural units: p (paragraph) / q (direct speech) / s (sentence)
 - p and s units have a “n” property encoding their number
- each lexical unit has the following properties:
 - word: the graphical form;
 - pos: the morphosyntactic tag;
 - col: the column number;
 - line: the line number.

4 Using TXM: commands

4.1 Describe corpus

For the selected corpus, that command displays a complete diagnostic of all the structural elements and their properties and of all the lexical units and their properties :

- number of words: the total number of lexical units of the corpus
- number of word properties: the number of available annotations for each word
 - for each annotation type: the name of the annotation and the total number of different values for this annotation, and some values
- number of structural units: the number of different structural units of the corpus
 - for each structural unit type: the name of the structure and the list of its attributes with their different values
 - for each structural unit attribute: the first elements of the list of values

Illustration 13 shows an example of corpus informations processed for the DISCOURS corpus.

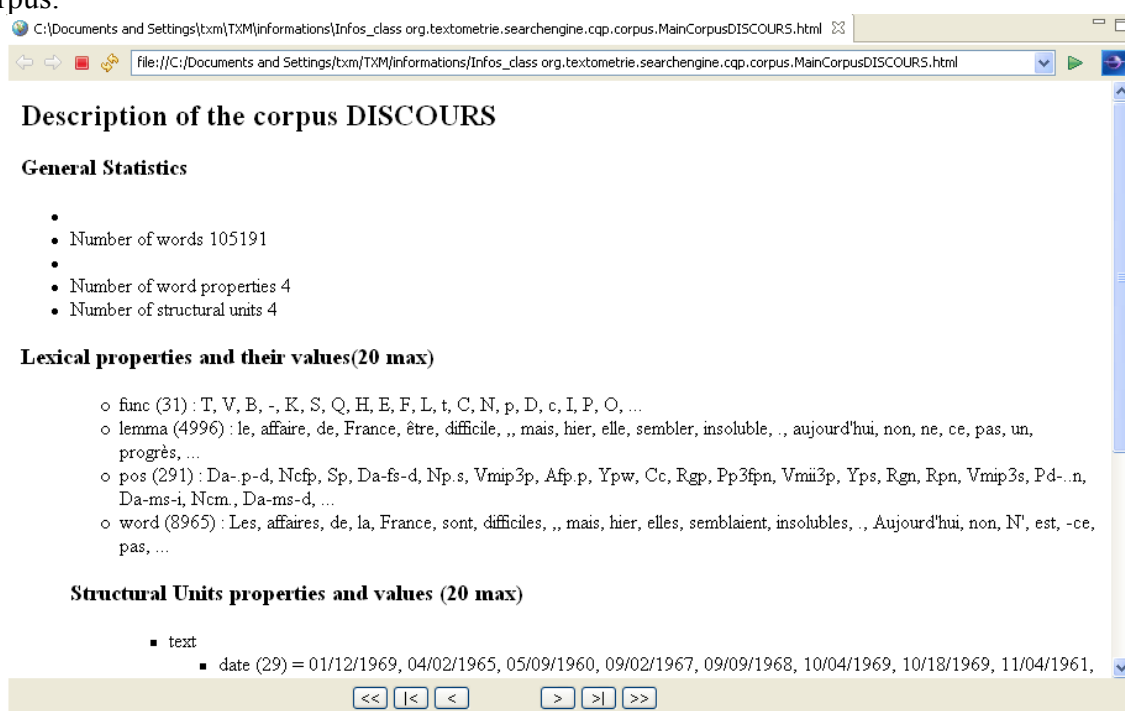


Illustration 13: DISCOURS Description

4.2 Read Edition

4.2.1 Corpus

For the selected corpus, that command displays the first page of the HTML edition of the first text of the corpus. The preamble of that edition presents all the metadata of the text.

In that edition, one can navigate:

- to the next '[>]' or previous '[<]' page;
- to the end '[>|]' or the beginning '[|<]' of the edition;
- to the next '[>>]' or previous '[<<]' text edition in the corpus order.

A double-click on a line of concordance (see below) opens or navigates directly to the page of an edition, while highlighting in red the selected keywords and in light red the other keywords of the concordance if they occur in the same page.

Illustration 14 presents the first page of the edition of the first text of the DISCOURS corpus :

- in that example, the metadata are : `id`, `file`, `loc`, `type`, `date`
 - `loc` : speaker name
 - `type` : type of speech
 - `date`
- each word has a flyover displaying its properties : `pos`, `func`, `lemma`
 - in that example, the mouse being over the word “équilibre”, the flyover displays:
 - `pos` = “Ncms” : common noun masculine singular (Multext tagset);
 - `func` = “-” : none
 - `lemma` = “équilibre”

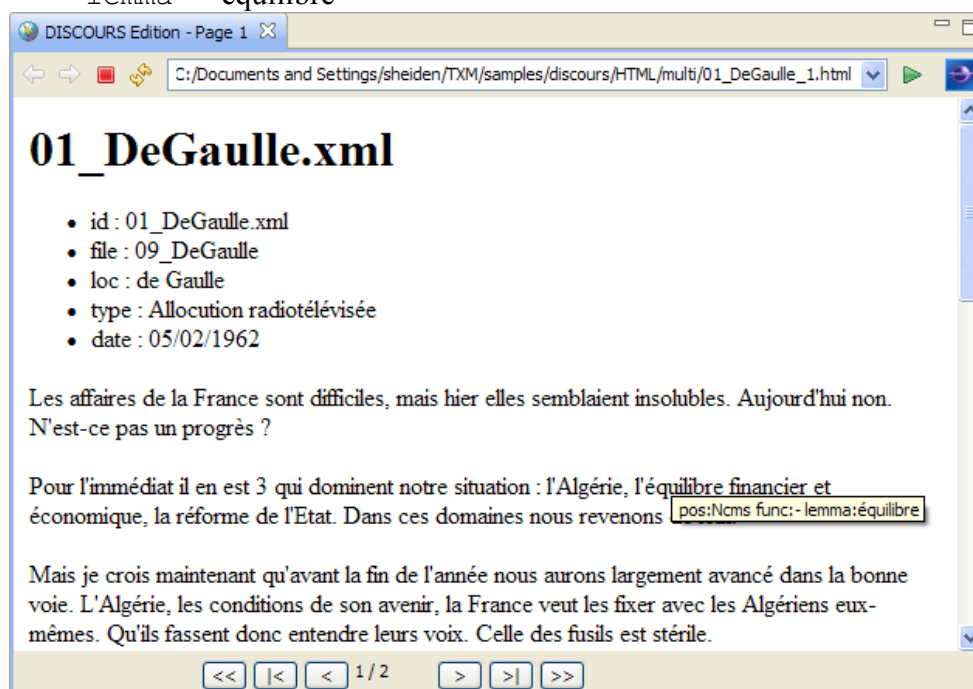


Illustration 14: DISCOURS Edition

4.2.2 Partition

The Text edition command for partitions allows to navigate into parts of the selected partitions in the explorer (see illustration 15).

The navigation system is similar to the system described above.

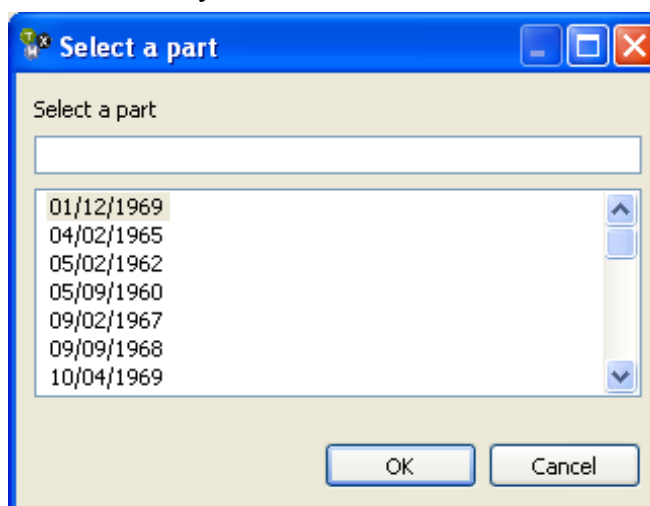


Illustration 15: Navigation window between the parts editions

4.3 Build Sub-corpus

That command is used to build a sub-corpus of the selected corpus. The new corpus is created as a child node in the corpus view.

That function opens a dialog box entitled “Create a sub-corpus”. It is composed of three tabs : one for simple sub-corpus build, one for assisted sub-corpus build and one for advanced sub-corpus build.

4.3.1 Simple sub-corpus building

Illustration 16 presents the sub-corpus builder simple tab form.

In that form, one has to:

- enter the name of the new corpus: the name displayed in the corpus view
- select a structural unit type
- select a property of that unit and its value

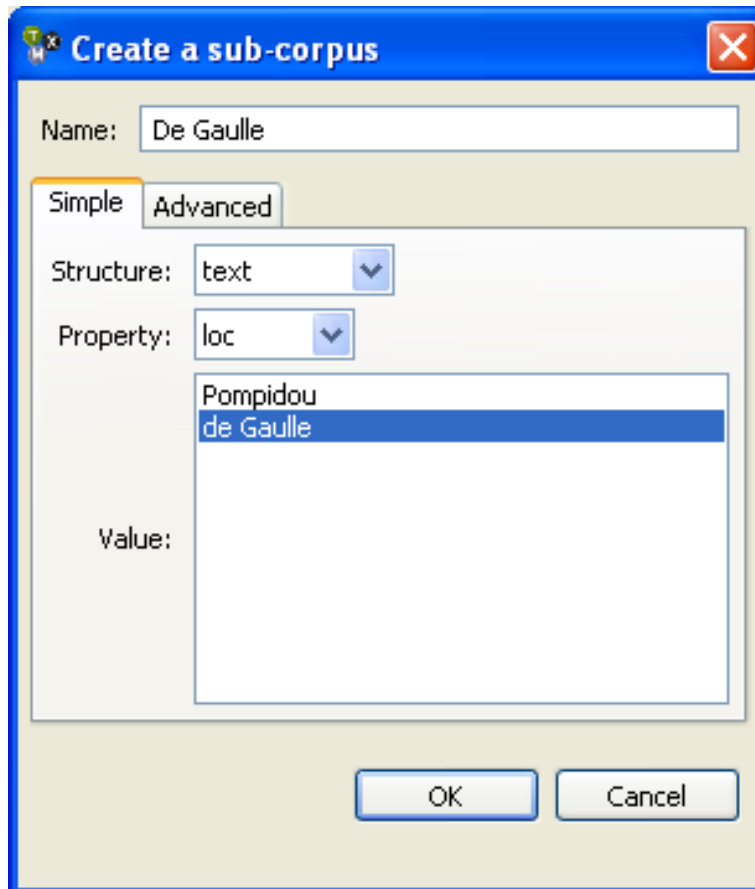


Illustration 16: Simple sub-corpus selection : build the sub-corpus of all the speeches of the De Gaulle president.

The new corpus will contain all the lexical units found in all the structural units of the given type with the given property set at the given value.

4.3.2 Assisted sub-corpus building

Illustration 17 presents the sub-corpus builder assisted tab form.

In that form, one can :

- Enter the name of the sub-corpus
- Check “all criteria” to treat all the criteria of the search or “some criteria” to treat some element constituting it.
- Select the structure of the sub-corpus
- Write the selection criteria :
 - add a criterion with the “+” button
 - delete a criterion with the “-” button
 - choose the property used by the criterion :
 - that contains or does not contain an property value
 - refresh the query of the sub-corpus
- Click on “OK” to create the sub-corpus

Illustration 17: Assisted sub-corpus selection : build a sub-corpus of the texts of the 12th century in verse.

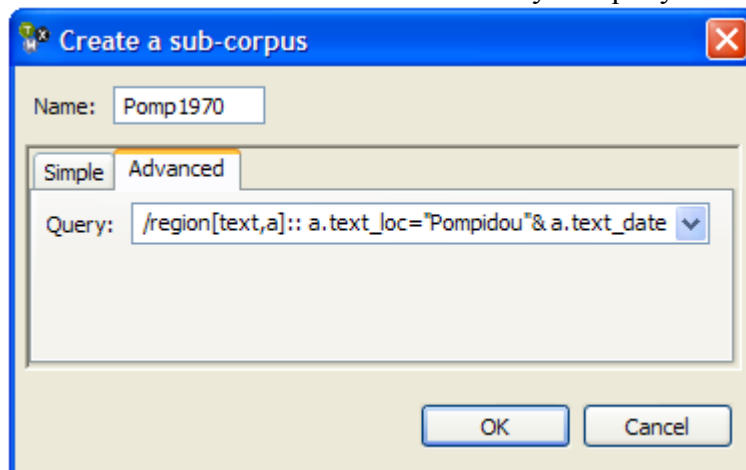
4.3.3 Advanced sub-corpus building

Illustration 18 presents the sub-corpus builder advanced tab form¹⁴.

In that form, one has to:

- enter the name of the new corpus : the name displayed in the corpus view
- write a CQP query which selects all the lexical units composing the sub-corpus

The new corpus will contain all the lexical units returned by the query.



4.4 Build Partition

That command is used to build a partition of the selected corpus. The new partition is created as a child node in the corpus view.

That function opens a dialog box entitled “Create Partition”. It is composed of three tabs : one for simple partition build, one for assisted partition build and one for advanced partition build.

4.4.1 Simple partition building

Illustration 19 presents the partition builder simple tab form.

In that form, one has to:

- enter the name of the new partition : the name displayed in the corpus view
- select a structural unit type
- select a property of that unit

The parts of the new partition will be built, as a sub-corpus, for each value of the selected property of the selected structural unit type. Parts can not be accessed individually, they can only be accessed as a whole through the partition object and contrastive commands like Specificity or Factorial Correspondence Analysis.

¹⁴ The complete expression is : `/region[text,a]:: a.text_loc="Pompidou"& a.text_date=".*1970"`

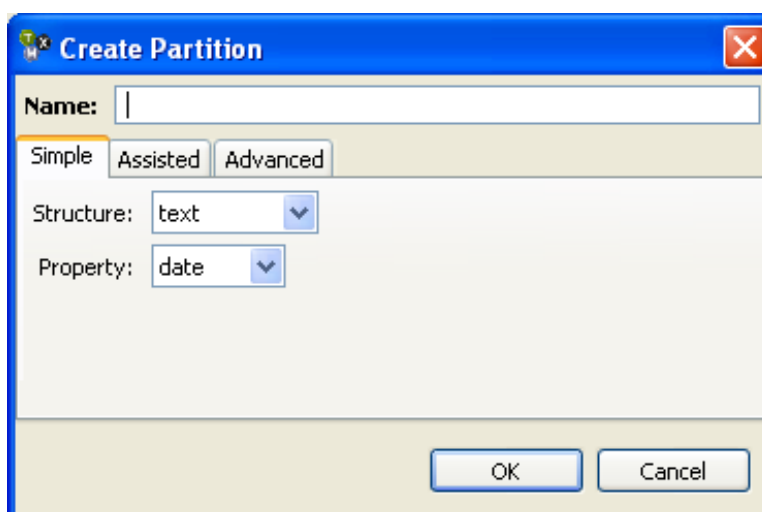


Illustration 19: Simple partition building : build a partition on every date of speech.

4.4.2 Partition building Assistant

Illustration 20 shows the assisted partition building window. Here, one can :

- enter the partition name which will be displayed in the Corpus View
- select a structural unit and its properties
- select the values that will compose a part of the partition
- click on “new part” to create an other part
 - enter the part title in the field
 - click on “Assign” to assign the selected values to the part
 - click on “Remove” to remove one or several values
 - click on the 'cross' to delete the part
- click on “Rmv all the parts” to delete with just one click all the parts
- click on “OK” to create the partition.

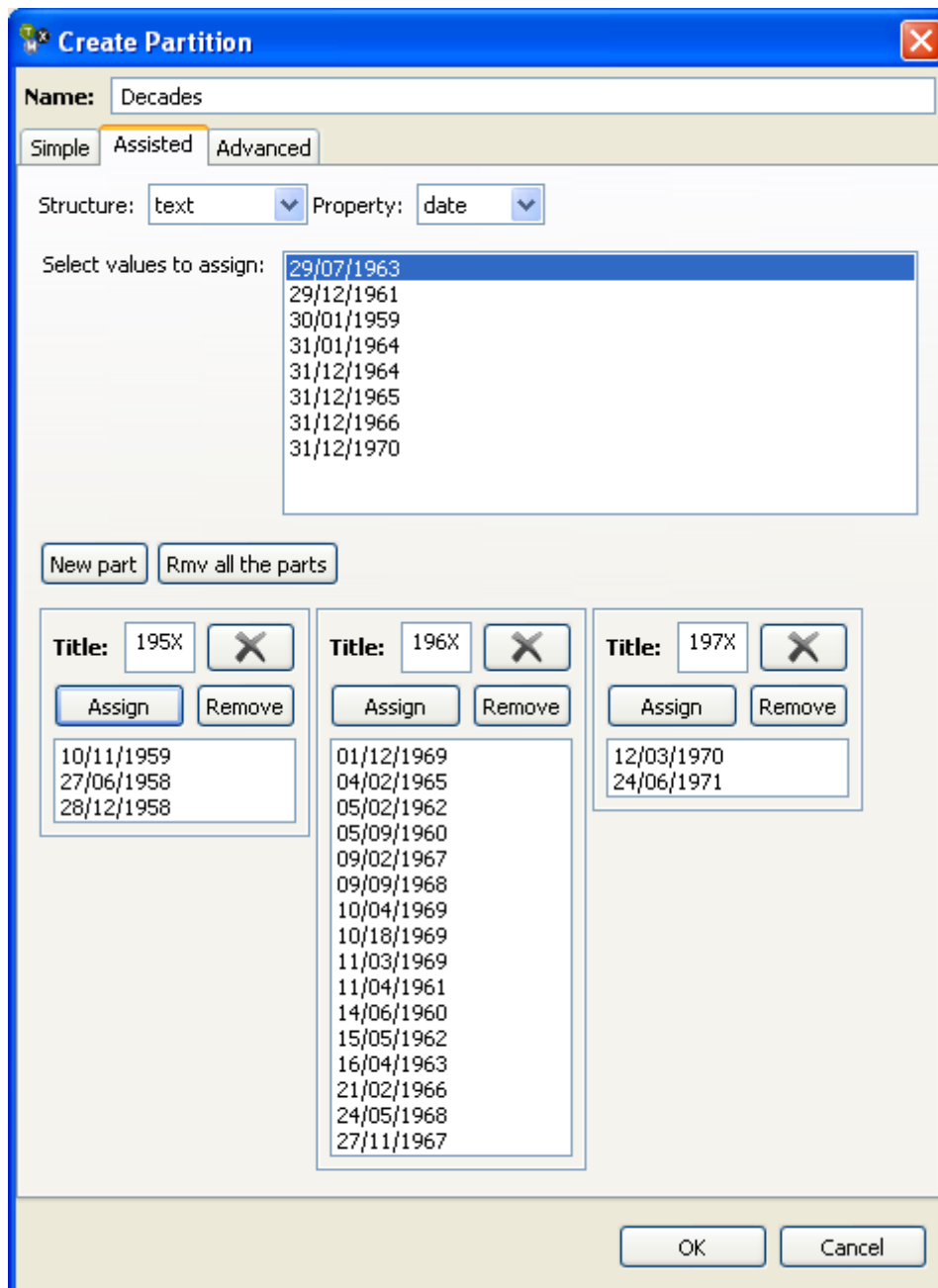


Illustration 20: Building a partition on the DISCOURS corpus with the text date values.

4.4.3 Advanced partition building

Illustration 21 presents the partition builder advanced tab form¹⁵.

In that form, one has to:

- enter the name of the new corpus : the name displayed in the corpus view
- write a CQP query which selects all the lexical units composing each part
 - use the '+' button to add a new part
 - use the '-' button to suppress a part

The new partition will be composed of all the parts defined, each one containing the lexical units returned by their respective query.

It is the responsibility of the user that all the parts sum to the whole corpus.

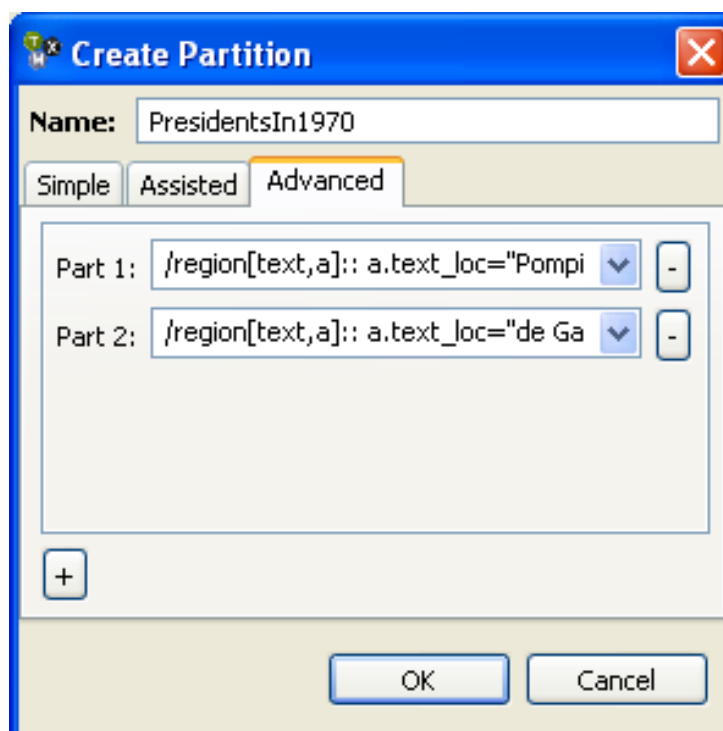


Illustration 21: Build a partition on every president for the year 1970.

4.5 Build Concordance

That command builds a kwic concordance of the search results of a specific CQP query expression on the selected corpus or sub-corpus.

¹⁵ The actual queries are :

```
- /region[text,a]:: a.text_loc="Pompidou"& a.text_date=".*1970"
- /region[text,a]:: a.text_loc="de Gaulle"& a.text_date=".*1970"
```

The initial search form is composed of:

- the CQP query input field;
- a button to access the history of queries;
- a button to access the lexical unit properties editor to select which properties will be displayed in the keyword column;
- the search button.

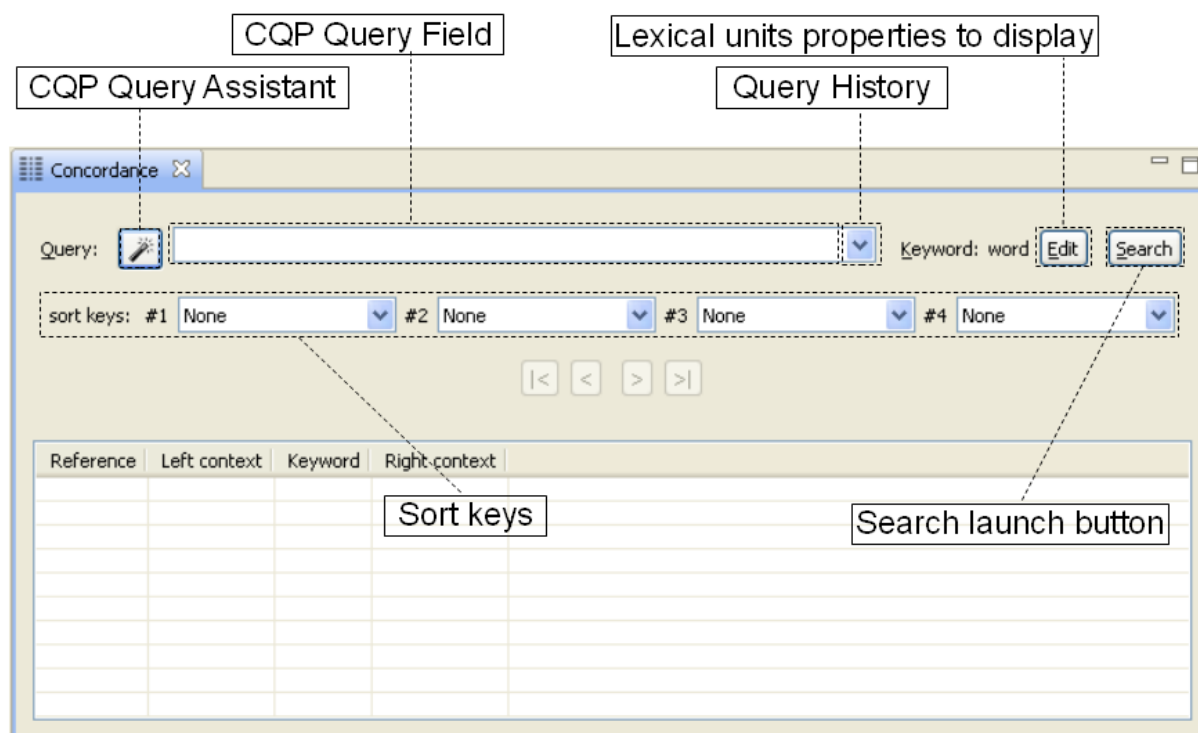


Illustration 22: Concordance Initial Search Form

4.5.1 Queries

The search engine allows you to express your queries in the CQP query language (see below section 5 “The Search Engine syntax”).

TXM defines a simplified syntax over the standard CQP queries to ease the writing of simple queries. For example: to just search for the “je” word (“I”, in French), you only need to write “je”, that is the two letters “j” followed by “e”, in the “Query” field.

For more elaborated queries, you have to conform to the CQP syntax. For example, to search for:

the “je” word followed by a verb

in the DISCOURS corpus, you can search for the query:

"je" [pos="V.*"]

That query can be decomposed as:

- the "je" part expresses the need for the "je" word to be there in the result;
- the [pos="V.*"] part expresses the need for a verb to be on the right of "je" next to the right:
 - the brackets [...] express the occurrence of just one lexical unit to be the next on the right of "je";
 - the pos="V.*" part expresses the constraint for that occurrence to have its pos property to match the "V.*" regular expression. In the DISCOURS corpus, which has been tagged by the Cordial tagger in the Multext tagset, this matches the pos property of all verbs (in that corpus, all the verbs have their pos property starting with "V").

An assistant is available to write queries. Click on the "Query Assistant" icon  and the following window will pop-up :

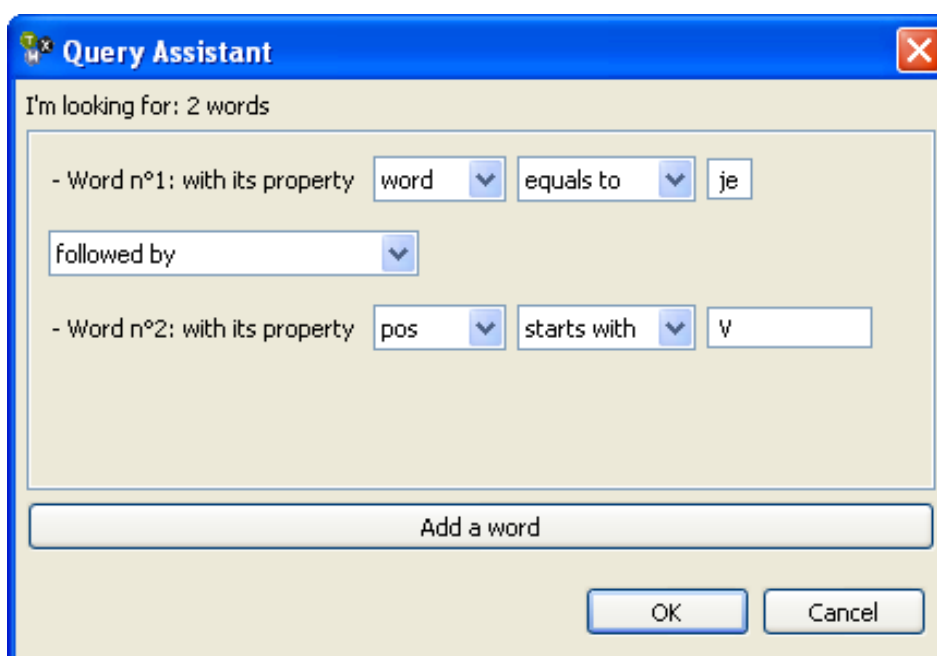


Illustration 23: Building a query for the word "je" followed by a verb.

- The button "add a word" adds a new word pattern to the query
- The first menu selects a word property
- The second menu defines the size of the search field
- The last field allows to tip letters or word
- The menu between two words allows to express if the words are consecutive or not.

If you validate with "OK", the query will appear in the query field.

The query is searched for by a click on the "Search" button.

TXM Reference Manual 0.5

Before drawing the concordance results, the console and the status line will notify you with the total number of matches.

The result is presented in illustration 24:

- there are 206 matches;
- it is the second page of the concordance which is displayed: from occurrences 22 to 41;
- the keyword column is composed of two consecutive words because of the query asking for the word “je” followed by a verb;
- the concordance is sorted alphabetically on the keyword column;
- the localization reference has been chosen to be the name of the speaker of the discourse in which the words occur;
- the contextual menu was opened (by a right click on the concordance):
 - Define references' pattern: to choose informations displayed in the Reference column;
 - Set Sort Property: to choose on which word property the sort will be done;
 - Multiple Sort: to select a sort on several keys;
 - Set contexts' size: to choose the number of words displayed in the contexts;
 - Select View properties: to choose which word properties will be displayed in each column.

This is the second page :
 Occurrences 1-21 on 206 matches

Reference	Left context	Keyword	Right context
text(loc):de Gaulle	étrangères d'Israël, que	je voyais	à Paris. Si Israël
text(loc):de Gaulle	amitié. Voilà ce que	je voulais	dire en ce qui concerne
text(loc):de Gaulle	complet. Voilà ce que	je voulais	vous dire, mesdames,
text(loc):de Gaulle	destin. Voilà ce que	je voulais	vous dire en commençant,
text(loc):de Gaulle	qui m'a demandé si	je voulais	mettre à l'ordre du
text(loc):de Gaulle	est confié. Mais si	je voulais	faire rire quelques-uns, ou
text(loc):Pompidou	citoyens. Voilà ce que	je voulais	vous dire.
text(loc):de Gaulle	de certains de ceux-là que	je voudrais	vous entretenir. Je crois
text(loc):de Gaulle	ai évoqué l'Europe,	je voudrais	que celui d'entre vous
text(loc):de Gaulle	avec quelque malice. Alors	je voudrais	vous dire très simplement comment
text(loc):Pompidou	la mesure du possible,	je voudrais	que ces entretiens soient une
text(loc):Pompidou	le passé. Néanmoins,	je voudrais	répondre un instant à ceux
text(loc):Pompidou	des intérêts pétroliers, et	je voudrais	vous rappeler que des deux
text(loc):Pompidou	éviter ou sanctionner. Mais	je voudrais	qu'on se représente ce
text(loc):Pompidou	dire, quelle tristesse quand	je vois	des Français accepter d'être
text(loc):de Gaulle	quelle idéologie ? Depuis que	je vis	, l'idéologie communiste a
text(loc):de Gaulle	, Françaises, Français,	je veux	vous dire que j'accepte
text(loc):de Gaulle	vous savez bien ce que	je veux	dire — qui obscurcissent plus
text(loc):de Gaulle	France, voilà ce dont	je veux	parler et vous exposer,
text(loc):de Gaulle	, et de nouveau,	je veux	montrer au pays où nous
text(loc):de Gaulle	votre attention. Cependant,	je veux	répondre à celui d'entre

The reference is build with the speaker name of each discourse

The keyword column is composed of two lexical units

Illustration 24: Concordance of the "je" word followed by a verb in the DISCOURS corpus.

4.5.2 Browsing

The concordance first displays the first page of results.

You can navigate through the results with the upper left panel buttons:

- "[|<]": go to the first page;
- "[<]": go to the previous page;
- "[>]": go to the next page;
- "[>|]": go to the last page.

The number of lines per page can be changed in the "File / Settings" menu, "TXM>User>concordances" preferences panel.

4.5.3 Returning to text

It is always possible to go back to the page of the edition containing the keyword occurrence by double-clicking on the corresponding line in the concordance.

The words composing the keyword are highlighted in red in the page, and keywords from other lines of the concordance occurring in the same page are highlighted in light red.

4.5.4 Sorting

You can sort the concordance by each column: "References", "Left context", "Keyword" and "Right Context" by clicking on their head line. You can change the sort order by clicking a second time. Default sort is on the word forms, but this can be changed in the contextual menu.

You can also do more complex sort, like sort on the right context then on the keyword. Select "Multiple sort" in the contextual menu to see the available sorts¹⁶.

4.5.5 Word properties displayed

You can choose which word properties, and in what order, will be displayed in each column.

There are two ways to do it :

- the current properties displayed for the keyword column are set under the query field. Press the "Edit" button to change the properties;
- in the contextual menu of the concordance, select the entry "Select view Properties"

4.5.6 References displayed

You can choose which informations will be displayed in the "Reference" column on the left side of each concordance line.

Selecting the "Define references' pattern" entry in the contextual menu (right click on the concordance) opens the dialog box of illustration 25:

¹⁶ This should be completely redesigned in the next release.

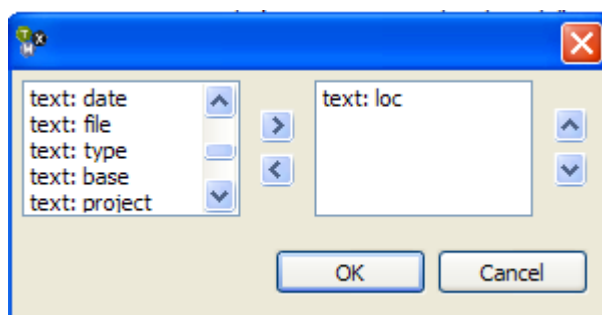


Illustration 25: Reference Pattern Dialog Box

The left panel lists all the properties of structural units and of lexical units.


For example, `text:loc` is the property “loc” of the structure “text”.

To choose a property, select it then click on the right ">" button to move it in the right panel which is the list of the properties which will be displayed in the reference column.

To unselect a property, select it in the right panel then click on the left "<" button to move it back to the left panel.

To change the order of properties in the right panel, use the up "^" and down "v" buttons.

4.5.7 Export

Concordances can be exported in the CSV format: select the concordance object in the corpus view and use to  icon in the toolbar or the "Export" entry in the contextual menu.

4.6 Cooccurrences

This command builds the table of the cooccurrents around a CQP query. The cooccurrence score¹⁷ allows you to sort cooccurrents according to their *a priori* encounter probability. The higher the score, the more surprisingly high is the number of observed encounters of the cooccurrent and the expression in the corpus.

The command opens a parameter window, like in illustration 26.

¹⁷P. Lafon, “Sur la variabilité de la fréquence des formes dans un corpus,” *Mots*, no. 1 (1980): 127-165.

Champ de requête CQP Seuils de fréquence

Assistant Propriétés des cooccurents Lancer la cooccurrence

Requête : j.* Calculer

Propriétés des cooccurents : word Editer Seuils : Fréq ≥ 1 Co-fréq ≥ 1 Indice ≥ 0.0

Contexte : mot structure s de - 50 à - 0 et de 0 à 50 inclure la structure du pivot dans les décomptes

Cooccurrent	Frequence	Cofrequence	Score	Distance moyenne
dis	28	28	13	20,9
Je	144	91	12	33,4
mon	61	47	11	29,8
vais	20	20	9	21,5
veux	20	20	9	25,7
répète	18	18	8	19,7
ma	28	24	7	28,1
questions	40	30	6	37,0
sais	14	14	6	30,2
moi	34	26	6	26,9
parler	20	17	5	18,5
répondre	22	18	5	24,8
Messieurs	17	15	5	30,2
voudrais	11	11	5	22,0
Vive	24	19	5	28,7
Mesdames	14	13	5	33,0
Françaises	19	16	5	29,2
dire	129	67	4	26,7
mes	13	12	4	25,0
parle	13	12	4	24,1
-	90	49	4	32,2
ce	712	291	4	31,0
poser	12	11	4	32,0

Taille du contexte Fréquence totale Score de spécificités

Liste des cooccurents Attraction des cooccurents

Distance entre le pivot et le cooccurrent

Illustration 26: Cooccurents of the words beginning by "j".

In this window, one can :

- Write a CQP query in the query field (or use the Query Assistant)
- Edit the cooccurents word properties
- Modify frequency thresholds to cut the results list. The cofrequency is the number of encounters of the cooccurrent and the CQP query occurrences in the corpus
- Choose a context size : if “structure” is selected, the right and left contexts can be set
- Sort the search results by clicking on the columns head.


To launch the search for the cooccurents, click on “compute”.

4.7 Lexicon and Index

The list of types (or any word properties) can be processed by two commands:

- Lexicon: computes the frequency list of all the values of a specific word property;
- Index: computes the frequency list of all the combinations of values of a specific list of properties for the result set of a specific CQP search query expression.

4.7.1 Lexicon

The Lexicon  command computes the complete frequency list of all the word forms, or pos tags, or word lemmas, etc. of a selected corpus or sub-corpus.

First, choose for which word property to build the list for:

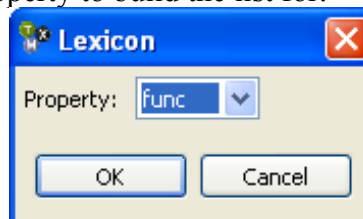


Illustration 27: Lexicon dialog box


The result is a sortable and exportable table:

word	F
,	8603
de	3916
.	2872
la	2672
et	2528
l'	2347
à	2095
les	1737
le	1674
que	1616
en	1525
qui	1335
des	1325
d'	1257
est	1236
qu'	1017
il	794
dans	757
pour	736

Illustration 28: word forms frequency list of the DISCOURS corpus sorted alphabetically.

You can sort each column by clicking on its header. Another click toggles the sort order. You can export this table into the CSV format.

4.7.2 Index

The Index  command computes the frequency list of the result set of a specific CQP search query expression, for a selected corpus or partition.

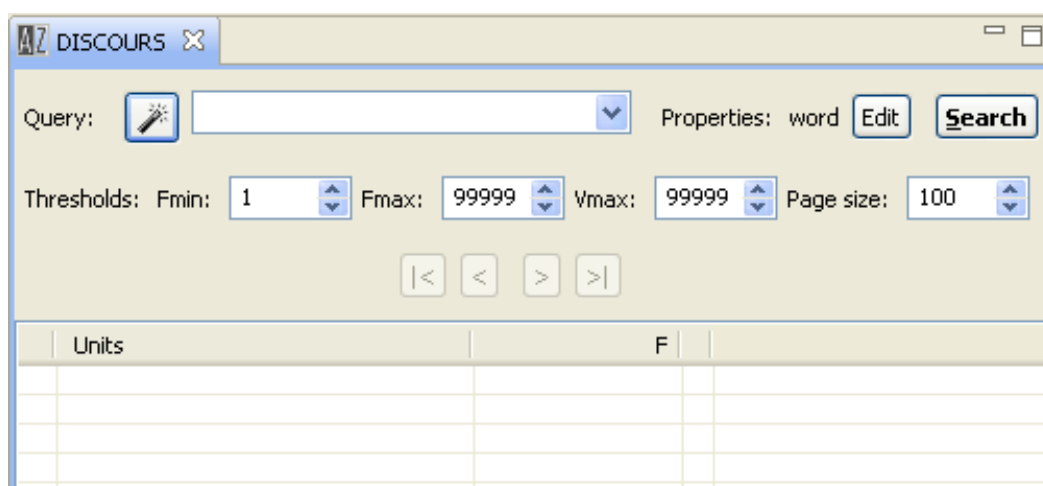


Illustration 29: Index initial dialog box.

4.7.2.1 Properties combination

First, select the combination of properties with the "Properties 'Edit'" button¹⁸:

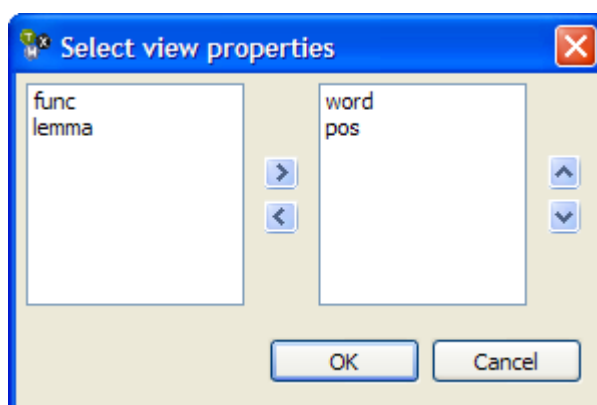


Illustration 30: Index word properties editor.

Select each property to combine in the left panel then use the arrows to move it to the right panel or to remove it:

- “>”: add the property to the combination (or double-click on the property in the left panel);
- “<”: remove the property from the combination (or double-click on the property in the right panel);
- “^”: display that property before in the combination (the top property in the right panel will be displayed first in the combination);
- “v”: display that property after in the combination.

¹⁸ In the example below, the 'word' property name stands for the graphical form of words.

4.7.2.2 Queries

You can write any CQP expression like in the concordance dialog box (or use the Query Assistant).

The screenshot shows the TXM Query Assistant window. The query is "[lemma='pouvoir']" and the properties are set to "word_pos". The search results are displayed in a table with two columns: "word_pos" and "F". The table lists various word forms and their frequencies.

word_pos	F
peut_Vmip3s	157
puisse_Vmsr3s	51
pouvoir_Ncms	35
pourrait_Vmcc3s	32
pu_Vmpasm	32
peuvent_Vmip3p	31
pouvoirs_Ncmp	26
pouvait_Vmii3s	20
pouvoir_Vmn--	18
puissent_Vmsr3p	17
pourraient_Vmcc3p	16
Pouvoirs_Ncmp	13
puis_Vmip1s	10
pourra_Vmif3s	10
pourront_Vmif3p	10
pouvons_Vmip1p	9
peux_Vmip1s	7
pouvaient_Vmii3p	5

Illustration 31: Index of the combination of the 'form' then 'pos' word properties for all the occurrences of the "pouvoir" lemma in the DISCOURS corpus.

4.7.2.3 Thresholds

You can limit the number of results with:

- Fmin: the minimum frequency necessary to be included in the list;
- Fmax: the maximum frequency allowed to be included in the list;
- Vmax: the maximum number of results to list;
- Page size: the number of results per page.

4.7.2.4 Browsing

The Index first displays the first page of results.

You can navigate through the results with the top buttons:

- "[|<]": go to the first page of results;


- "[<]": go to the previous page;
- "[>]": go to the next page;
- "[>|]": go to the last page.

4.7.2.5 Hypertext

The Index command is linked to the Concordance command.

Select some lines in the Index results with the mouse¹⁹, then in the contextual menu (Ctrl-click) choose "Send to concordance": a corresponding query will be generated for a new concordance to build.

4.8 Specificities

The Specificity command  uses a probabilistic model²⁰ to compute the overused (and the underused) word properties (word forms, lemmas, pos...) of each part of a partition (for example, of each text or of each century of the corpus) or of a sub-corpus with respect to its parent corpus.

4.8.1 Partition specificities

The partition is associated to a structural unit and to one of its properties of which each possible value is associated to a part in the partition.

The Specificity command opens the following parameters dialog box:

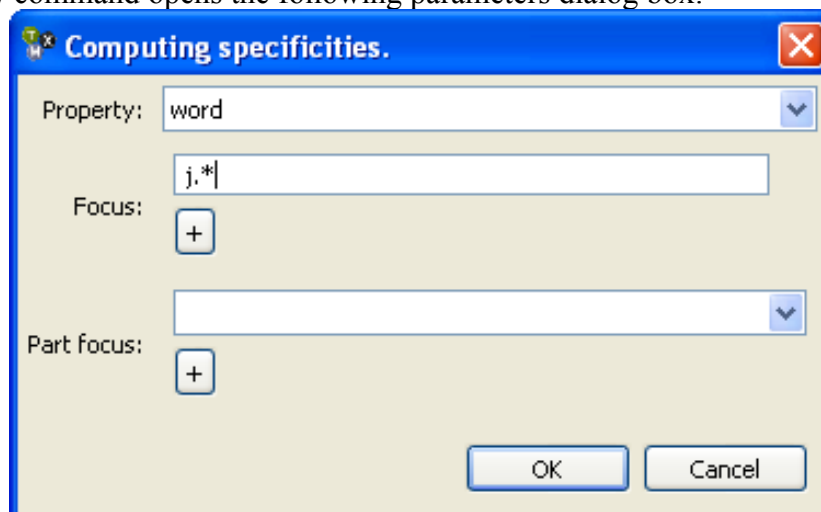


Illustration 32: Specificity for a partition dialog box.

The parameters are the following:

- Word property: you can choose the word property the command will be applied on;

¹⁹ Shift-click selects several contiguous lines. Ctrl-click selects several non-contiguous lines.

²⁰Ibid. <http://www.persee.fr/web/revues/home/prescript/article/mots_0243-6450_1980_num_1_1_1008>. [originally presented at the Association for Literary and Linguistic Computing conference at Oxford the 4-5th of April 1976]

- Property filter: you can filter the values of the word property (lines) that will be processed with a regular expression (this is not a CQP expression [yet]);
 - you can add as many filters as you need with the “+” button;
 - if you don't specify any filter, all the values will be processed. For example, for the word form property, all the word forms will be processed;
- Part filter: you can filter the values of the structural unit property (columns) that will be displayed. You can use the “v” button to access the available values;
 - you can add as many filters as you need with the “+” button;
 - if you don't specify any filter, all the parts will be considered by the command.

The result is a table with:

- lines: the word property values appearing in all parts;
- columns: the values of the structural unit property taken into account – the parts;
 - the first column gives the total frequency of the word property value in the corpus. 'T' is the total number of words ;
 - the other columns gives the logarithm of the specificity score of the word property value in the specific part. 't' is the total number of words in the part.

Illustration 33 presents the Specificity scores of all the word forms matching “j.*” (that is starting with a 'j' character) in the partition of discourse “type”s for the DISCOURS corpus. The table is sorted on the score of the part for the “Allocution radiotélévisée” type, decreasing.

Units	Frequency T=105191	Allocution radiotélévisée t=49868	Conférence de presse t=41834	Entretien radiotélévisé t=13489
je	359	2.5707	-2.6872	0.3344
jeune	7	1.3264	-1.5413	-0.4172
jamais	68	0.8234	-1.2965	0.6008
jeunesse	20	0.7375	-0.5905	0.3165
journalistes	2	0.6483	-0.4404	0.6197
jeunes	13	0.3723	0.378	-0.3106
janvier	9	0.3601	-0.3112	-0.5364
juin	5	0.3454	0.5038	0.3041

Illustration 33: Specificity of "j." word forms in the discourse type partition of the DISCOURS corpus.*

4.8.1.1 Sorting

You can sort the table by columns by clicking on their head line. You can change the sort order by clicking a second time.

When a score column is sorted downward, the top words are considered overused in the corresponding part with respect to the whole corpus, the last words are considered underused and the middle words – around the zero score - are considered commonplace (the score is useless for them).

4.8.1.2 Graphics

The scores can be visualized graphically.

Select some lines in the results table with the mouse²¹, then in the contextual menu (right-click) choose “Graphic”:

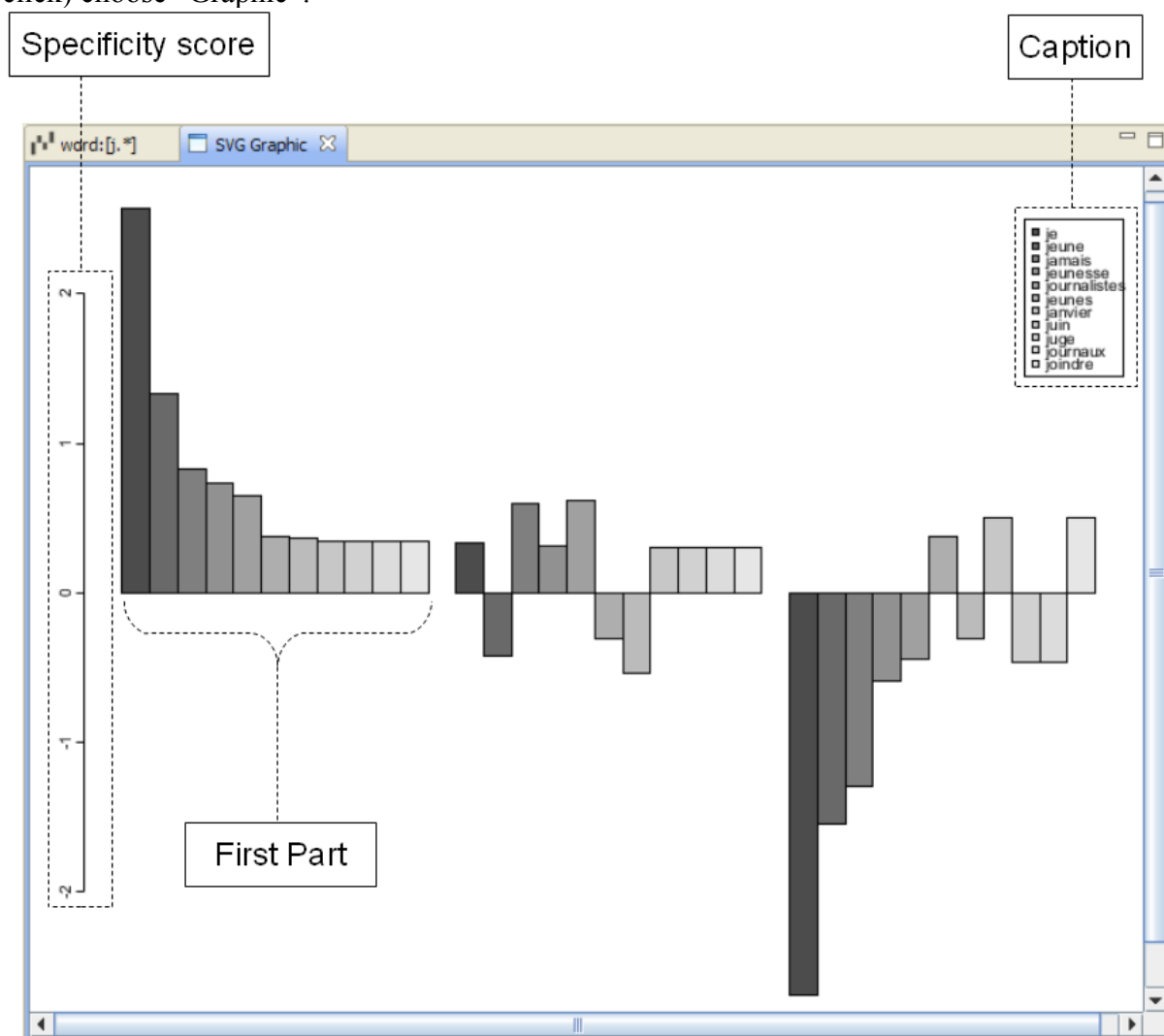


Illustration 34: Specificity graphic of the "je", "jeune"... word forms between discourse genres in the DISCOURS corpus.

In the graphic:

- each part will be represented by a set of contiguous bars in the same order as in the table;
- for each word property value selected (the word form in the example) the score will be represented by a bar of the same color in each part;
- the legend in the upper right of the graphic gives the key of colors for each value.

²¹ Shift-click selects several contiguous lines. Ctrl-click selects several non-contiguous lines.

4.8.1.3 Browsing the graphic

To ease the reading of the graphic, you can:

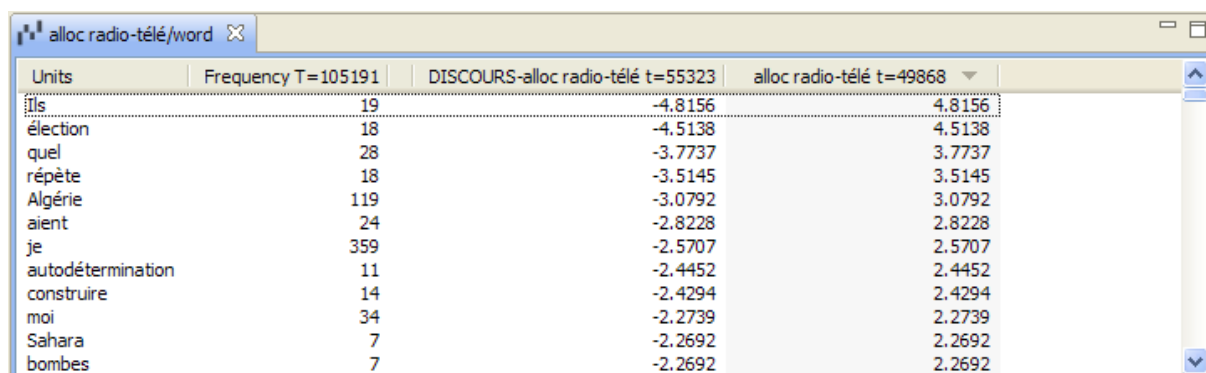
- pan: with Shift + Left mouse button + drag
- zoom in: Shift + Right mouse button + drag
- zoom to selection: Ctrl + Left mouse button + drag
- rotate: Ctrl + Right mouse button + drag
- reset the view: F5

4.8.2 Sub-corpus specificities

For a sub-corpus, that command allows you to choose on which word property to compute the specificity scores on. Thus, the command opens the same dialog box as the Lexicon command in illustration 27.

The command then displays, after the columns of the word property values and their global frequency, two lists of scores:

- one list for the score in the complement of the sub-corpus with respect to the parent corpus (named “corpus name - part name”);
- one list for the score in the sub-corpus with respect to the parent corpus (named “part name”).



Units	Frequency T=105191	DISCOURS-alloc radio-télé t=55323	alloc radio-télé t=49868
ils	19	-4.8156	4.8156
élection	18	-4.5138	4.5138
quel	28	-3.7737	3.7737
répète	18	-3.5145	3.5145
Algérie	119	-3.0792	3.0792
aient	24	-2.8228	2.8228
je	359	-2.5707	2.5707
autodétermination	11	-2.4452	2.4452
construire	14	-2.4294	2.4294
moi	34	-2.2739	2.2739
Sahara	7	-2.2692	2.2692
bombes	7	-2.2692	2.2692

Illustration 35: Specificity scores of the word forms of the "Allocution radiotélévisée" discourse genre in the DISCOURS corpus.

4.9 Progression

A progression displays graphically the evolution of one or more patterns throughout the corpus. This command is launched on a corpus. It makes a cumulative or density graphic and adds the selected structure limits in the corpus. When launching the Progression command, a parameters window is opened, like in illustration 36. Then, you can :

- Choose the progression display type : cumulative or by density
- Choose the structural unit displayed (each vertical bar corresponds to a unit limit) and one of its property to display
- Filter property values with a regular expression
- Add one or more CQP queries to display (possibly with the Query Assistant) with the “add” button. You can also remove one query with the “delete” button.

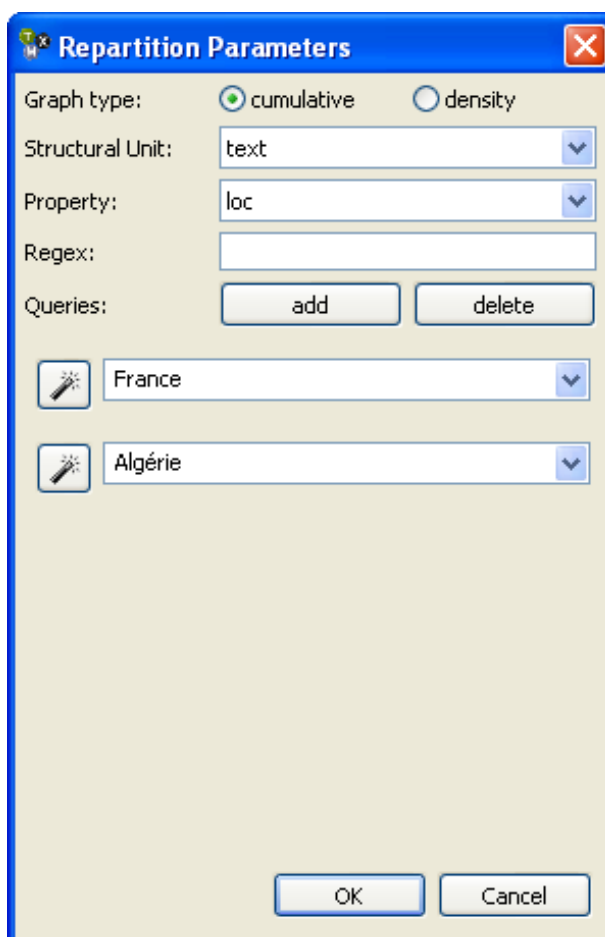


Illustration 36: Progression processing parameters for the "France" and "Algérie" words, in the DISCOURS corpus.

Clicking on “OK” displays a progression graphic such as in illustration 37. In this graphic, the speaker name is displayed at the beginning of each discourse. The curves represents the progression of the “France” and “Algérie” words.

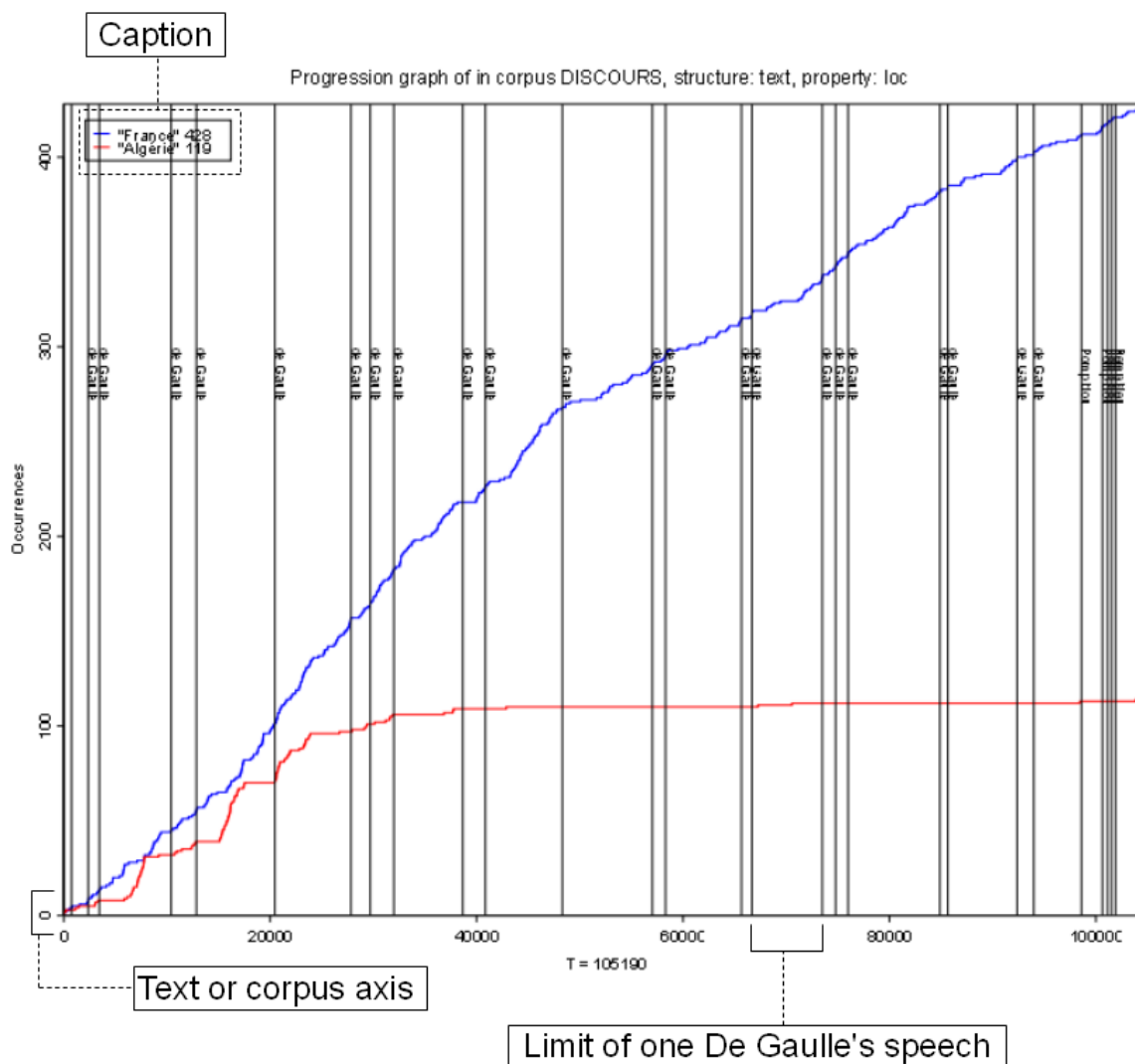



Illustration 37: Progression graphic on the "France" and "Algérie" words in the DISCOURS corpus.

The graphic can be exported with the “Export” button in the toolbar.

4.10 Correspondence Analysis

The CA command  computes the correspondence factor analysis algorithm²² on a partition based on the frequency of values of one of their word properties (word forms, lemmas, pos...) in each part.

²²Jean-Paul Benzécri et al., *L'analyse des correspondances* (Paris: Dunod, 1973). Computed by the “CA” R package.

TXM Reference Manual 0.5

Applied on a partition (of at least four parts) or on a lexical table, the command first allows you to choose on which word property to compute the algorithm. Thus, the command opens the same dialog box as the Lexicon command in illustration 27.

The results are then presented in two different windows:

- the first one displays the first factorial plane graphic
- the second one displays the factorial analysis parameters for the graphic interpretation. It is divided into four tabs :
 - singular values
 - lines information
 - columns information
 - barplot graph of the singular values.

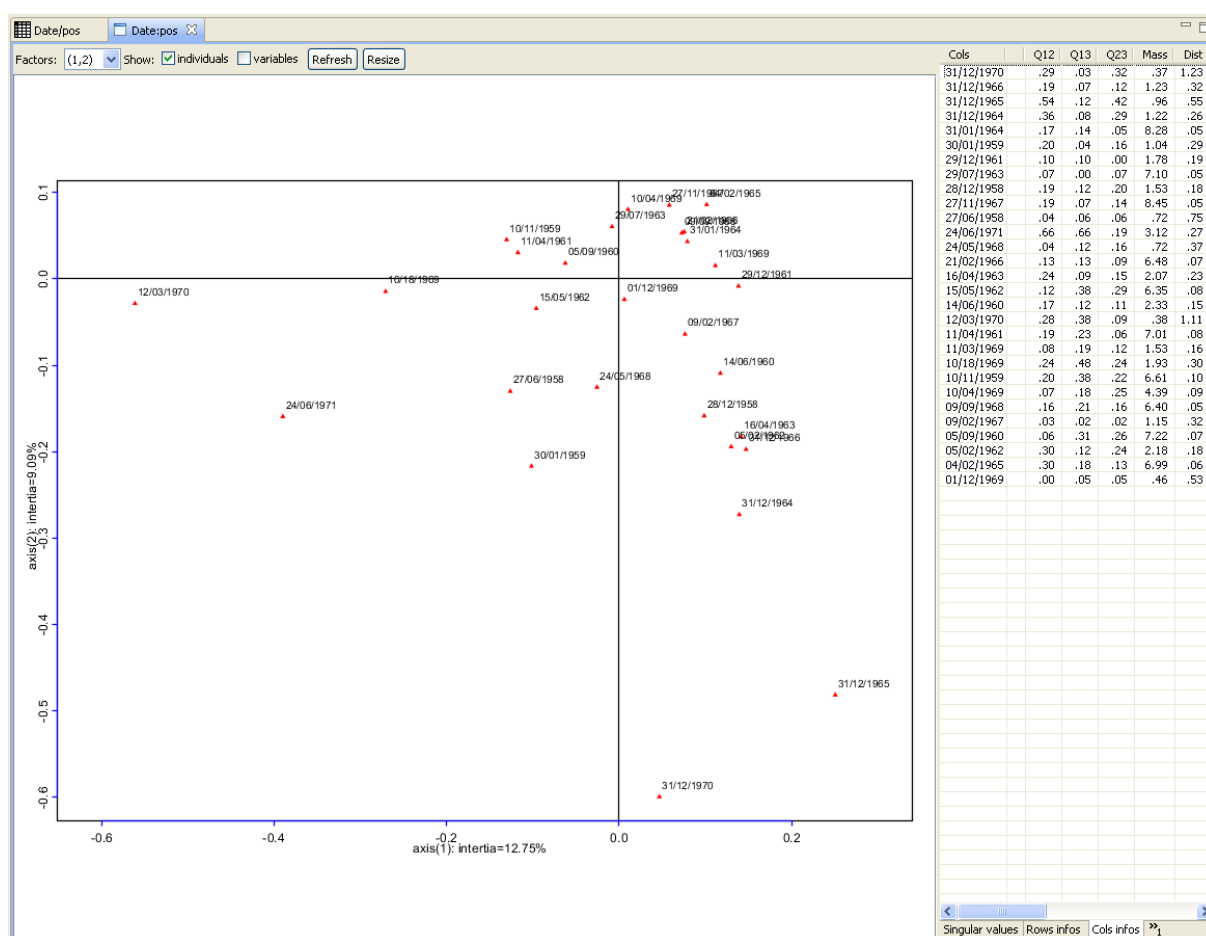


Illustration 38: Graphics obtains from a lexical table, with the “Date” property, on the DISCOURS corpus.

The CA window can display the individuals or the variables or both : for that, check or uncheck “individuals” and “variables”, then update the view by clicking on “refresh”. The graphic can be resized by clicking on “Resize”(see also the graphics shortcuts in section 6.2, for zooming, pan, rotation, etc.)

By default, the correspondence analysis plot shows only the parts in the plane.

You can change this in the CA preference page:

- “Show individuals in graph”: display word property values;
- “Show variables in graph”: display parts.

In the right pane, many details information are available for variables, individuals and singular values reading.

For each singular values, the table displays the value numbers, the singular values and the percentage of the singular value.

Display of lines and columns tables :

- Quality of the plane : for each plane, the quality of the representation of the point is computed as the sum of the point's \cos^2 values on the two axis defining the plane. The closer the quality is to 1, the less is distorted the point position after its projection onto the plan.
- Relative weight : frequency divided by the sum of the other words' frequency (lines).
- Distance of the point from the origin (that is the center of the representation or the center of the cloud of points)
- Contribution of the point to the axis building. Contributions sum to 100, and points with the highest contributions are used to interpret the axis.
- \cos^2 of the point along each axis : a measure of the angle between the vector representing the point and the axis. A \cos^2 close to 1 indicates a well represented point on the axis, a \cos^2 close to 0 indicates that the projection of the point on the axis is highly distorted (the point coordinate on that axis should not be considered to compare the point' position with other points). A point with a small \cos^2 on both axis, for a specific plane, has a misleading position in the representation ; its apparent proximity to other points should not be interpreted in this plane.
- Point coordinates.

4.11 Lexical table

The lexical table displays the frequency of the lexical units of a partition.

This table can be created from a partition or from an index of a partition. Once the partition selected, choose a word property for the table to create, like in illustration 39 :



Illustration 39: Lexical table property selection.

TXM Reference Manual 0.5

Here is the description of the table content : one entry by line, one part by column. This table total can be edited, lines and columns can be merged or deleted. It is also possible to filter the number of lines or to choose lines to keep by a minimal frequency threshold.

The CA or Specificities command create automatically a lexical table.

Information on results : total, frequency, number of lines

Number of lines and minimal frequency edition

Table edition

	Freq	29/12/1961	31/12/1970	24/06/1971	11/04/1961	09/02/1967	10/10/1969	16/04/1963	31/12/1964	24/05/1968	31/12/1965	31/12/1966	05/09/1960	04/11/1960
Da...	1928	33	11	40	153	20	25	44	18	15	8	14	155	
Nlfp	2061	35	5	48	154	21	48	45	17	14	31	20	135	
Sp	12...	245	52	329	864	148	235	245	186	88	150	163	907	
Da...	2677	47	9	77	172	39	55	51	31	26	20	38	192	
Np.s	732	10	5	9	71	7	5	9	6	3	5	8	59	
Vm...	938	17	2	26	78	9	12	20	5	4	12	9	102	
Afp.p	252	5	2	7	17	2	5	4	0	1	2	5	16	
Ypw	8888	147	40	308	565	117	187	232	131	70	97	112	609	
Cc	3609	75	12	111	247	65	80	93	45	30	38	46	279	
Rgp	4104	62	10	129	275	44	85	80	55	31	35	43	331	
Pp...	72	0	0	3	6	4	2	1	0	0	0	0	9	
Vm...	189	1	0	5	14	4	0	2	0	0	1	0	15	
Yps	3405	56	17	113	256	39	55	80	44	28	28	47	232	
Rgn	1226	17	15	56	106	8	21	33	13	14	9	9	93	
Rpn	983	18	2	35	103	9	17	27	8	3	9	1	80	
Vm...	3117	67	7	124	209	37	77	75	43	27	18	34	228	
Pd...	596	8	0	32	41	5	29	9	4	4	6	5	55	
Da...	630	13	1	23	51	5	23	11	5	5	5	9	34	
Ncm	1437	38	4	39	79	16	14	37	37	13	17	18	111	
Da...	4956	88	15	156	303	51	87	87	64	35	32	61	289	
Ncms	4917	82	16	142	320	56	105	89	74	34	40	65	245	
Pp...	1085	20	1	45	79	7	17	26	9	8	5	8	88	
Pp...	453	4	1	24	34	1	9	4	4	3	4	5	44	
Mc...	443	13	2	10	22	5	5	7	8	0	5	3	33	
Pr...	1165	22	5	36	96	14	12	25	12	6	12	11	90	
Ds...	349	9	5	4	25	1	5	18	11	7	12	11	6	
Nlfs	5952	114	24	182	355	75	116	122	72	46	62	68	393	
Nlfc	785	13	3	36	58	12	6	14	4	1	6	20	73	
Af...	1728	35	4	43	107	14	43	36	21	12	10	22	100	
Np...	385	6	0	5	35	3	3	4	3	2	1	2	28	
Dd...	183	2	2	6	22	1	7	1	0	0	0	1	25	
Nlcmp	2289	44	6	45	162	17	29	49	26	18	29	36	175	
Pp...	520	2	9	28	38	7	1	14	12	2	20	12	48	
Vm...	300	1	7	19	17	5	1	9	9	1	11	9	32	
Pp...	651	3	3	67	86	3	32	1	6	8	1	2	46	
Vm...	423	1	3	44	47	2	25	1	6	5	1	2	32	
Cs	2377	36	4	91	186	30	54	44	21	11	28	29	188	
Val...	4	0	0	0	0	0	0	0	0	0	0	0	0	
Vm...	1405	25	4	44	117	12	20	29	16	5	13	15	99	

Word property

Part of the « date » partition

Illustration 40: Lexical table on the "Date" partition of the DISCOURS corpus.

In the above illustration, the lexical table is created from the "Date" partition. One can :

- define the total number of lines and a minimal frequency threshold. The "Keep" button applies the parameters to the current table.
- merge or delete columns : clicking on the "merge or delete columns" button opens a values selector (see illustration 41) :

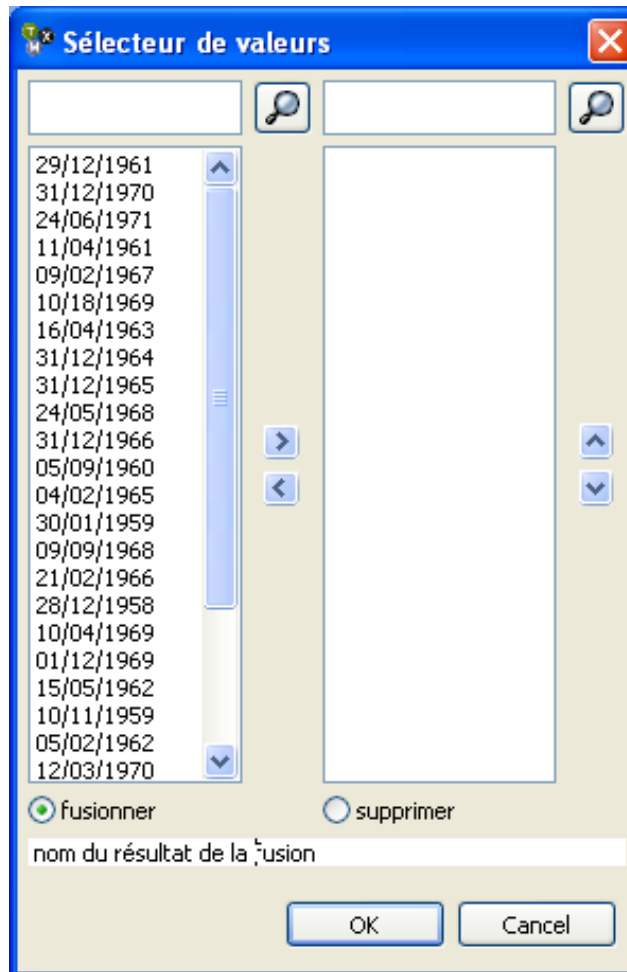


Illustration 41: Columns selection window.

- This window allows you to select several columns. Use the search field (to filter select columns by a regular expression) or select directly a value column or several ones.
- “>” adds a value
- “<” removes a value
- Then, check “merge” or “delete” to select the operation to apply (you can give a name to the merge result)
- merge or delete lines :
 - click on the “merge or delete lines” button : a dialog box similar to that window above allows you to edit the number of lines to keep.
 - or select directly lines in the table, right-click to delete or merge the selected lines
 - click on “OK” to refresh the table
- export the table from the contextual menu
- sort columns by clicking on their heads.

4.12 TXM settings

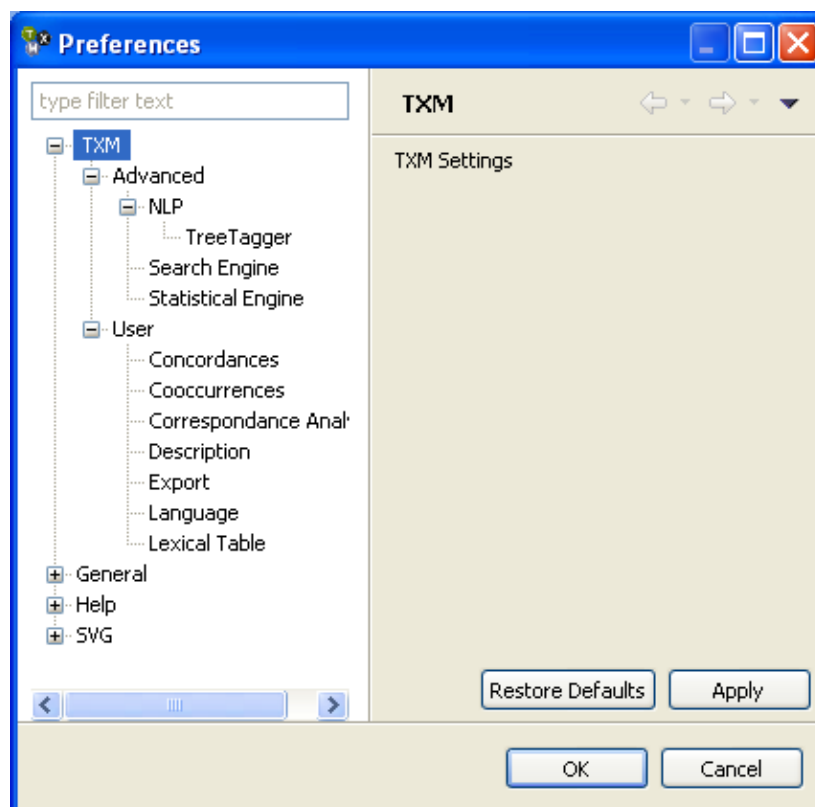


Illustration 42: TXM settings window.

- **Advanced** : advanced settings of the TXM platform
 - **NLP** : software settings for the Natural Language Processing tools
 - **TreeTagger** : morphosyntactic tagger used by TXM
 - **Search Engine** : parameters of the CWB server integrated into TXM.
 - **Statistical Engine** : parameters of the statistical engine R integrated into TXM.
- **User** : default settings for all TXM commands
 - **Concordances** : number of lines per page, context size
 - **Cooccurrences** : minimal frequency, maximum number of cooccurrents, minimal score
 - **CA** : show the individuals or the variables in graphics, change columns format (it uses the specifications of the Java class : `DecimalFormater`. For more information, see: <http://download.oracle.com/javase/1.4.2/docs/api/java/text/DecimalFormat.html>)
 - **Description** : the number of property values to display
 - **Export** : encoding format of the results export
 - **Language** : English or French language interface.

4.13 Commands relationship

COMMANDS	FROM	TO	USED BY
CA	Partition Lexical Table		
Concordances	Corpus		Cooccurrences
Cooccurrences	Corpus	Concordances	
Corpus	Corpus		Cooccurrences Concordances Corpus Description Index Lexicon Partition Progression Text Edition
Description	Corpus		
Index	Corpus Partition	Concordances Progression	Lexical Table of a Partition
Lexical Table	Partition Partition index		CA Specificities
Lexicon	Corpus	Concordances Progression	
Partition	Corpus		CA Specificities Lexical Table Text Edition
Progression	Corpus		
Specificities	Partition Lexical Table		
Sub-corpus	Corpus		Corpus commands + Specificities
Text Edition	Corpus Sub-corpus Partition		

5 The Search Engine syntax

5.1 Quick introduction

All the queries you write in the “Query” fields of the Concordance and Index commands to express their focus, are given to the internal TXM search engine for resolution. Those queries must obey the CQP²³ language syntax and semantics. Here is an elementary introduction to it:

- to search for a simple word, just cite it literally:

la

[a wrapper will finalize the query to "la", which is the right query, for you]

- to make the search not case sensitive add the “%c” modifier :

"la"%c

[modifiers are always written outside double quotes]

- to make the search not diacritic sensitive add the “%d” modifier :

"la"%d

[you can combine the “c” and “d” modifiers together in “%cd”]

- to search for a compound word, put it in double quotes:

"parce que"

[the fact that word tokens contains blanks depends on the tokenizer used to import the corpus into TXM. See below to look for all the words containing blanks]

- to search for a word beginning by “l”, write:

l.*

[“.” means <any character>, “*” means <possibly repeat the last expression, which is – any character – here>.

The result is thus <any sequence of characters, including none>. Those special meaning characters are called “operators” or “jokers”. They can appear anywhere in a query but with a specific syntax. If you want to express a particular operator character literally in a query, use the “\” operator immediately before it.]

- to search for a word ending by “a”, write:

.*a

- to search for a word ending by “a”, possibly with a “s” after, write:

²³ For “Corpus Query Processor”: from the IMS Open Corpus Workbench technology (<http://cwb.sourceforge.net>).

TXM Reference Manual 0.5

`. *as?`

[“?” means possibly the last expression, which is “s” here]

- to search for a word beginning by “l” and ending by “a”, write:

`l.*a`

- to search for a word containing the letter “l”, write:

`.*l.*`

- to search for a word containing a blank, write:

`" .* . *"`

[blanks have no meaning in CQP expressions except in double quotes]

- to search for a word beginning with “L” or “l”, write:

`[Ll].*`

[the “[...]” construction means <one of the following characters can match, and just one>]

- to search for a word beginning with a lowercase, write:

`[a-z].*`

[the “-” in the “[a-z]” construction means <a value of character between the “a” character to the “z” character can match>, that is <any lowercase character, and just one>]

- to search for two adjacent words, write:

`"le" "jour"`

[please note that the blank character in the middle of the query is part of the CQP query language and is not a literal blank. It can, for example, be repeated without changing the meaning of the query]

- to search for three adjacent words, write:

`"le" "jour" "où" (etc.)`

- to search for a word which is a verb, that is whose part-of-speech property (called “pos” in the sample corpora) value is beginning with “V”, write:

`[pos="V.*"]`

[1] this is true for the sample corpora of TXM. Values of properties of words depend on the annotations that have been performed on the corpus in the import process into TXM. Morphosyntactic taggers produce different tagsets so you have to read their documentation to craft the right query for a specific tagset. 2) Please note that the “[...]” in that query are not the same, and don’t have the same meaning, as the previous ones. The previous ones were implicitly enclosed in double quotes. Here “[...]” means <the expression inside the square brackets concerns exactly and only one word>]

TXM Reference Manual 0.5

- to search for a verb at the imperfect tense, write:

```
[pos="V..i.*"]
```

[only true for the “Multext” tagset of the sample corpora]

- to search for a verb followed by a noun, write:

```
[pos="V.*"] [pos="N.*"]
```

- to search for the word “je” (I) followed by a verb, write:

```
"je" [pos="V.*"]
```

[in fact, this query is equivalent to: [word="je"] [pos="V.*"]]

- to search for the word “je” followed by a verb, with one word in between, write:

```
"je" [] [pos="V.*"]
```

[here, the “[]” expression means <a word without any constraint, that is any word>]

- to search for the word “je” followed by a verb, possibly with one or two words in between, write:

```
"je" []{0,2} [pos="V.*"]
```

[the “{}” modifier adds the capacity to count how many elements must match]

- to search for the word “je” followed by a verb at any distance but not crossing sentence boundaries, write:

```
"je" []* [pos="V.*"] within s
```

[1) the “within” close expresses a constraint on the boundaries of all structural units. 2) please note that the first “*” operator (counting from left) has not the same semantics as the second one (which is the same as the ones we have introduced before, that is <repeat “.”>). The first “*” means <repeat the “[..]” expression before (which is a word occurrence expression – and not a character occurrence expression)>. Summary: “*” outside double quotes repeats word expressions on their left, “*” inside double quotes repeat character expressions on their left.]

- to search for the word “je” followed by the verb “aimer” at any distance but not crossing paragraph boundaries, write:

```
"je" []* [lem="aimer"] within p
```

To understanding all the level of CQL queries, you can read the “Reference manual of CQL expressions” : <http://weblex.ens-lsh.fr/doc/weblex/refregexpcqp.html>
Please see the “CQP User's Manual” for a complete description at <http://www.ims.uni-stuttgart.de/projekte/CorpusWorkbench/CQPUserManual/HTML/>

6 Driving the TXM platform with scripts

6.1 Running Groovy scripts and commands

The ability to script the TXM platform gives the end user the opportunity to automatically:

- call any TXM commands: search a CQP expression with the search engine, compute a statistical model score with the statistics engine, export and save results in a file, etc.
- use different parameter values for those commands;
- record and reproduce a set of commands for a regular analysis.

It is also a way for the end user to extend the platform with new commands²⁴.

Scripts are written in the Groovy scripting language (<http://groovy.codehaus.org>).

You will find a short introduction to the language at :

<http://onjava.com/pub/a/onjava/2004/09/29/groovy.html>

At least three books will also introduce you to the language:

- *Groovy in action*²⁵
- *Groovy programming: an introduction for Java developers*²⁶
- *Programming Groovy : dynamic productivity for the Java developer*²⁷

The text of the scripts to execute can be stored in a file or simply selected and copied from an editor window (see the “Text Editor” section).

The best way to start writing your own Groovy script is first to modify the sample scripts released with TXM in the “C:\Documents and Settings\\TXM\scripts” directory²⁸. For example the “conc.groovy”²⁹ script which computes a concordance of the “je” word in the DISCOURS corpus and then exports and saves it in the “conc.txt” text file.

To do so, use the “File View” (see the “File view and text editors” in section 3.2.1.1.2) to find, open and change the script, for example by changing the word searched for and the name of the backup file, and then execute it through the contextual menu of the text editor (accessed by a right click of the mouse).

²⁴ In the same way as you extend MS Word by a Visual Basic macro.

²⁵Dierk König et al., *Groovy in action* (Greenwich: Manning, 2007).

²⁶Kenneth A. Barclay et W. J. Savage, *Groovy programming: an introduction for Java developers* (Morgan Kaufmann Publishers, 2007).

²⁷Subramaniam Venkat, *Programming Groovy: dynamic productivity for the Java developer*, Pragmatic Bookshelf. (Raleigh: Daniel H. Steinberg ed., 2008).

²⁸ Please note that no security policy has been enforced on Groovy scripts in the TXM platform for the moment, so be vigilant with script code of which you don't know the provenance.

²⁹ You can also read that script on line at

“<http://textometrie.svn.sourceforge.net/viewvc/textometrie/trunk/Toolbox/0.4.7/org.textometrie.toolbox/src/groovy/org/textometrie/test/conc.groovy?revision=1080&view=markup>”

The best reference documentation for all the available TXM commands and their parameters is the Java documentation of the TXM platform at <http://textometrie.sourceforge.net/javadoc/index.html>.

For example, the parameters of the “Concordance” class constructor are described in the “`java.org.textometrie.functions.concordances`” package documentation for the “Concordance” class, that is at

“<http://textometrie.sourceforge.net/javadoc/index.html?java/org/textometrie/functions/concordances/Concordance.html>”.

All classes and methods described in that documentation are available for a Groovy script.

6.2 Running R scripts and commands

The TXM platform uses the R statistical environment to implement some statistical models. To this end, it loads specific packages, processes results and displays them in its user interface. For example, it displays in a new window the specificity barplot graphics computed by R.

This version of TXM allows you to also edit and run yourself R scripts from within its user interface.

The text of the scripts to execute can be stored in a file or simply selected and copied from an editor window (see the “Text Editor” section).

The best way to start writing your own R script is first to modify the sample scripts released with TXM in the “`C:\Documents and Settings\<your login name>\TXM\scripts`” directory.

For example:

- The « `sample.R` » script generates a vector of points following a normal law, then displays them;
- The « `HelloWorldR.groovy` » script shows how to embed a R script inside a Groovy script and then to call it;
- For scripts generating graphics, the « `executeRscript.groovy` » script shows how to call the « `plot100.R` » R script from Groovy while allowing the graphic to be displayed inside the TXM user interface windows.

7 Import modules

The import modules available in the RCP version of TXM are stored in the « `scripts/import` » subdirectory of the TXM home directory (`~/TXM`). Currently, only the main launch script for each module is available to the user (the files named “`xxxLoader.groovy`”)³⁰.

7.1 Clipboard module

7.1.1 input

That module imports the raw text copied in the system clipboard. The “`lb`” property is added to each word to encode the line number.

7.1.2 output

As output, a unique text structure (`text`) encloses words segmented by separator characters.

7.1.3 annotation

Morphosyntactic description and lemma properties are added to each word by the TreeTagger software.

7.1.4 edition

The text is edited by taking care of spaces and punctuations marks between words, and is paginated by blocs of `n` words.

7.2 XML-TEI BFM module

7.2.1 input

That module imports the files encoded in the XML-TEI P5 BFM format of the source directory.

The input format is defined by the encoding manual of the Medieval French Base project - Base de Français Médiéval (BFM). It is based on the XML TEI P5 format to encode the text body and metadata.

For further information, please see:

- The BFM XML-TEI encoding manual: http://bfm.ens-lyon.fr/article.php3?id_article=158 (in French)

³⁰ Because of an unresolved bug, see : https://listes.cru.fr/wiki/txm-users/public/retours_de_bugs_logiciel#synthese_des_retours_de_bugs

- The Text Encoding Initiative consortium: <http://www.tei-c.org>

7.2.2 annotation

A morphosyntactic description is added to each word by TreeTagger using the old French linguistic model “rgaqcj.par”. The tagset used by this model is CATTEX2009 (see http://bfm.ens-lyon.fr/article.php3?id_article=176).

7.2.3 edition

Text edition type is close to the one produced in the « Queste del Saint Graal » project (see <http://textometrie.risc.cnrs.fr/txm>). However, that component of the module will be later replaced by the XSLT+CSS stylesheets of Alexei Lavrentiev to get similar and maintained results.

7.3 XML-TXM module

7.3.1 input

That module imports the files encoded in the XML-TXM UTF-8 format (extension '.xml') of the source directory. It doesn't do any tokenization of words because the XML-TXM³¹ format already encodes them with “<w>” tags.

One interest of that format is that it requires little work to be imported into TXM. Although not finalized yet, it is always compatible with the TEI encoding scheme. There is one text per XML file.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<TEI xmlns="http://www.tei-c.org/ns/1.0"
xmlns:txm="http://textometrie.org/1.0">
  <teiHeader type="text">
    <fileDesc>
      <titleStmt>
        <title>Grec essai</title>
        <respStmt>
          <resp id="ucl">initial tagging</resp>
        </respStmt>
      </titleStmt>
    </fileDesc>
    <encodingDesc>
      <classDecl>
        <taxonomy id="lemma"><bibl type="tagset"/></taxonom
y>
        <taxonomy id="pos"><bibl type="tagset"/></taxonomy>
```

³¹ The XML-TXM format is defined as a XML-TEI P5 extension specifically for the TXM platform.

```

my>
        <taxonomy id="intext"><bibl type="tagset"/></taxono
my>
        </classDecl>
        </encodingDesc>
</teiHeader>
<text id="grec-try-1">
    <w id="w_1">
        <txm:form>mot</txm:form>
        <interp resp="#resp" type="#lemma">lemme</interp>
        <interp resp="#resp" type="#pos">pos</interp>
        <interp resp="#resp" type="#autre">autre</interp>
    </w>
    <!--... -->
</text>
</TEI>

```

7.3.2 output

Each XML tag level generates one structural level. The properties of words are imported from the content of the “<interp>” sub-elements of each “<w>” element.

7.3.3 annotation

No annotation is added by this module.

7.3.4 edition

Each text is edited by taking care of spaces and punctuations marks between words, and is paginated by blocs of n words.

7.4 XML/w module

7.4.1 input

That module imports the XML files found in the source directory.

The “<text>” tag is reserved for this module. Any “<text>” tag found in the source will be renamed “<textunit>” by the module.

If some words are delimited by “<w>” tags, they will be taken as such with their properties imported from the tag attributes. Care must be taken so that all “<w>” elements have the same number and names of attributes.

7.4.2 output

Each XML tag level generates one structural level.

7.4.3 edition

Each text is edited by taking care of spaces and punctuation marks between words, and is paginated by blocs of n words.

7.5 Transcriber+CSV module

7.5.1 input

Body of text

That module imports the transcription files encoded in the XML-TRS (extension “.trs”) format found in the source directory. That format is generated by the Transcriber software³².

The files must come with the “trans-14.dtd” file to be valid.

Each transcription will be associated to one textual unit, or text.

Text metadata

Text metadata are imported from a file encoded in the CSV format, called “metadata.csv” and found in the same directory as the sources.

The column separator is the comma “,”, the field character³³ is the double-quote ‘”’.

The first header line names each metadata column.

The first column must be named “id”, the following ones can be named freely but without using any accented or special characters.

The first column must contain the name of the source file (without the extension) corresponding to the metadata of the line.

The metadata will be injected at the level of each transcription, if present.

Parameters

That module uses a parameter file called “import.properties” coming with the transcription files.

With it, one can set three different parameters:

- `removeInterviewer`: can be “true” or “false” if the module should ignore the content of the speech of each interviewers in the import process;
- `metadataList`: the list of metadata to be considered. Metadata are separated by a “|” character;
- `csvHeaderNumber`: the number of header lines in the metadata CSV file.
 - 1 = there are only metadata identifiers;
 - 2 = there is one line of identifiers and one line of long identifiers;
 - 3 = there is one line of identifiers, one line of long identifiers and one line of metadata types³⁴.

³² See <http://trans.sourceforge.net/en/presentation.php>

³³ The field character surrounds column data containing commas or spaces, etc.

³⁴ That last value is not used in that version of the software.

7.5.2 output

The structure of Transcriber files is reproduced:

- each Transcriber section corresponds to a `div`³⁵ structure;
- a speech turn corresponds to a `sp` structure;
- an speech utterance corresponds to a `u` structure.

The two kind of Transcriber events are managed:

1. milestones: comments, short noise...
2. word segments: pronunciation, uncertainty...

Milestone events comments are encoded in the `event` property of the following word.

For events surrounding several words, the event descriptions are concatenated in the `event` property of the words transcribed between the “begin” and the “end” Transcriber events.

Some metadata are copied at the word level (`spk`) and others at some structural levels (`u@spkattrs`, `textAttr@<metadata>`, `div@topic@endtime@starttime@type`, `sp@speaker@endtime@starttime@overlap`, `event@type@desc`) to help sub-corpus building.

7.5.3 annotation

Morphosyntactic description and lemma properties are added to each word by the TreeTagger software³⁶.

7.5.4 edition

The edition reproduces to the HTML edition of Transcriber. The table of metadata values is edited at the beginning of each transcription. Each transcription is paginated every `n` words after a speech turn. Events and comments are enclosed in parentheses. Synchronization information is edited between brackets.

7.6 Hyperbase module

7.6.1 input

That module imports files encoded in the old Hyperbase format in the source directory. That is, with the following text delimiting line:

```
...
&&& Long text name, TextName, ShortTextName &&&
```

```
...
```

Page break lines (encoded by “\§”) are interpreted. They are encoded as `p` structures.

³⁵ `div`, `sp` and `u` elements are loosely adapted from the TEI standard.

³⁶ Mind that TreeTagger linguistic models are built from written text corpora: tagging results on orthographic transcriptions must be checked.

7.6.2 annotation

Morphosyntactic description and lemma properties are added to each word by the TreeTagger software.

7.6.3 edition

Each text is edited by taking care of spaces and punctuations marks between words, and is paginated by blocs of n words.

7.7 Alceste module

7.7.1 input

That module imports text encoded in the Alceste software format. Which is nearly raw text with some escape characters.

There are two ways to delimit a text:

1. a line of the form: 0001 &Attr1 Val1 &Attr2 Val2... &AttrN ValN
2. a line of the form: **** &Attr1 Val1 &Attr2 Val2... &AttrN ValN

To encode a compound word, one can replace the spaces between words by a “_” character. For example, “l'assemblée nationale” can be segmented into two words: “l'” and “assemblée_nationale”.

The Alceste format allows also one to encode speech turns, but that module doesn't manage that encoding.

7.7.2 output

As output, a text structure (`text`) encloses words segmented by separator characters.

7.7.3 annotation

Morphosyntactic description and lemma properties are added to each word by the TreeTagger software.

7.7.4 edition

Each text is edited by taking care of spaces and punctuations marks between words, and is paginated by blocs of n words.

7.8 CNR+CSV module

7.8.1 input

Text body

That module imports files encoded in the CNR format from the source directory. The CNR format is produced by the Cordial software and corresponds to a TSV format with the tabulation character as column separator and no field character.

The CNR columns define respectively:

- para: the paragraph number;
- sent: the sentence number;
- form: the graphical form of a lexical unit;
- lem: the lemma;
- pos: the part-of-speech or morphosyntactic description;
- func: the syntactic function.

Text metadata

Text metadata are imported from a file encoded in the CSV format, called “metadata.csv” and found in the same directory as the sources.

The column separator is the comma “,”, the field character is the double-quote ‘”’.

The first header line names each metadata column.

The first column must be named “id”, the following ones can be named freely but without using any accented or special characters.

The first column must contain the name of the source file (without the extension) corresponding to the metadata of the line.

7.8.2 output

As output, texts are structured by text (`text`), paragraphs (`p`) and sentences (`s`).

Word properties are directly imported from the CNR column values.

7.8.3 annotation

No annotation is added by this module.

7.8.4 edition

Each text is edited by taking care of spaces and punctuations marks between words, and is paginated by blocs of `n` words. The table of metadata values is edited at the beginning of the first page.

7.9 TXT+CSV module

7.9.1 input

Text body

That module imports raw text files found in the source directory (extension “.txt”).

The “lb” property is added to each word to encode the line number.

Text metadata

Text metadata are imported from a file encoded in the CSV format, called “metadata.csv” and found in the same directory as the sources.

The column separator is the comma “,”, the field character is the double-quote ‘”’.

The first header line names each metadata column.

The first column must be named “id”, the following ones can be named freely but without using any accented or special characters.

The first column must contain the name of the source file (without the extension) corresponding to the metadata of the line.

7.9.2 output

As output, each textual unit (`text`) is built with properties imported from the metadata file, and encloses words segmented by separator characters.

7.9.3 annotation

Morphosyntactic description and lemma properties are added to each word by the TreeTagger software.

7.9.4 edition

The text is edited by taking care of spaces and punctuation marks between words, and is paginated by blocs of n words. The table of metadata values is edited at the beginning of the first page.

8 Keyboard Shortcuts

8.1 Text Editor

Command	Shortcut
Help	
Show Key Assist	Ctrl+Shift+L
Selection	
Select All	Ctrl+A
Select Line Start	Shift+Home
Select Line End	Shift+End
Select Next Word	Ctrl+Shift+Right
Select Previous Word	Ctrl+Shift+Left
Edit	
Copy	Ctrl+C, Ctrl+Insert
Paste	Ctrl+V, Shift+Insert
Cut	Ctrl+X, Shift+Delete
Delete	Delete
Undo	Ctrl+Z
Redo	Ctrl+Y
To Upper Case	Ctrl+Shift+X
To Lower Case	Ctrl+Shift+Y
Find	
Find and Replace	Ctrl+F
Find Next	Ctrl+K
Find Previous	Ctrl+Shift+K
Incremental Find	Ctrl+J
Incremental Find Reverse	Ctrl+Shift+J

Move

Text Start	Ctrl+Home
Text End	Ctrl+End
Line Start	Home
Line End	End
Next Word	Ctrl+Right
Previous Word	Ctrl+Left

Go to Line	Ctrl+L
Last Edit Location	Ctrl+Q

Delete

Delete Line	Ctrl+D
Delete to End of Line	Ctrl+Shift+Delete
Delete Next Word	Ctrl+Delete
Delete Previous Word	Ctrl+Backspace

Move line

Move Lines Up	Alt+Up
Move Lines Down	Alt+Down

Insert line

Insert Line Above Current Line	Ctrl+Shift+Enter
Insert Line Below Current Line	Shift+Enter

Other

Join Lines	Ctrl+Alt+J
Scroll Line Up	Ctrl+Up
Scroll Line Down	Ctrl+Down
Duplicate Lines	Ctrl+Alt+Up

Copy Lines Ctrl+Alt+Down

Toggle Folding Ctrl+Numpad_Divide

Mode

Toggle Insert Mode Ctrl+Shift+Insert

Toggle Overwrite Insert

Toggle Block Selection Alt+Shift+A

Quick Diff Toggle Ctrl+Shift+Q

Show Ruler Context Menu Ctrl+F10

File

New Ctrl+N

Save Ctrl+S

Close Ctrl+W, Ctrl+F4

Close All Ctrl+Shift+W

Print Ctrl+P

Properties Alt+Enter

Refresh F5

Misc

Word Completion *Alt+/'*

8.2 Graphics Output

Pan Shift+Left Mouse+drag

Zoom in&out Shift+Right Mouse+drag

Zoom to selection Ctrl+Left Mouse+drag

Rotate Ctrl+Right Mouse+drag

Reset the view F5

8.3 Windows

Editor Windows

Next Editor	Ctrl+F6
Previous Editor	Ctrl+Shift+F6
Quick Switch Editor	Ctrl+E
Switch to Editor	Ctrl+Shift+E
Show System Menu	Alt+-

View

Maximize Active View or Editor	Ctrl+M
Next View	Ctrl+F7
Previous View	Ctrl+Shift+F7
Show View Menu	Ctrl+F10
Show Key Assist	Ctrl+Shift+L
Show View	Alt+Shift+Q, Q
Show View (View: Console)	Alt+Shift+Q, C

9 TXM Glossary

Categories:

- com : Command
- mod : Data Model
- fmt : File Format
- int : Interface
- nlp : Natural Language Processing
- exp : Search Query expression
- soft : Software
- met : Textometry Methodology

Entry	Cat	Description
AFR	nlp	the standard code for the 'old French' language.
Alceste	soft	a commercial software of textometry.
annotation	mod	a unit property (lexical or structural) from a logical point of view.
CATTEX2009	nlp	a morphosyntactic tagset for the old French language.
character	mod	the elementary component of word forms.
clipboard	mod	a component of the operating system where a selection of text can be stored by the 'Copy' command.
ClipN	int	all the corpora created from the clipboard are automatically named 'Clip'+<a number>.
CNR	fmt	the data format of the output file of Cordial.
command	com	an elementary action available in TXM.
concordance	com	a way to present the results of the search engine where every hit is displayed on its own line with some contextual words around.
console	int	TXM displays various messages while executing commands in a special window called the 'console'.
Cordial	nlp	a commercial tagger.

TXM Reference Manual 0.5

corpus	mod	a compilation of word sequences. Sequences come from texts, in whole or in part. Root corpora are build from a selection of texts
CQL	exp	for <Corpus Query Language>, query language managed by CQP, applied to corpus.
CQP	soft	for <Corpus Query Processor> software component processing the search queries to build the index, concordances, etc.
CSV	fmt	for <Comma Separated Values>, a textual file format where each record is separated by a newline and where each property, or value, is separated by a chosen character (like comma).
Ctrl	int	the 'Ctrl' or 'Control' key on the keyboard.
directory	mod	a file containing other files or directories on the file system of the user. A directory can be designated by a path.
document	mod	a text from a logical point of view.
editor	com	a textual window in which the text can be modified, like a source text file or a script file.
encoding	mod	the way in which an information is represented in a source corpus.
export	com	the action of saving in a file the results of a TXM command for external processing crediting.
factorial correspondence analysis	com	the action of reducing the dimensionality of a [parts x words] matrix according to the correspondence analysis algorithm. The new dimensions are represented by eigenvectors called factors. The parts and the words from the original matrix can be displayed in the resulting factorial planes.
file	mod	an elementary container of information on the user file system : like a text or a corpus source. A file can be designated with a path.
flyover	int	a small popup window displayed while the mouse moves over an object in the interface, for example a word in an edition.
focus	int	a way to concentrate a command on a specific word event, for example through a search query.

TXM Reference Manual 0.5

form	mod	the graphical form of a word, generally computed by tokenizers.
frequency	met	the total number of occurrences of an event (a word occurrence, a sequence of words occurrence, etc.) in a corpus.
Groovy	soft	the computer language in which the TXM platform scripts are written.
HTML	fmt	the data format of web pages.
Hyperbase	soft	an academic software of textometry.
import	mod	the process of integrating into the platform a corpus from its source files.
index	com	the action of listing word property combinations with their frequency for the occurrences of a search query.
index	soft	file built by TXM to accelerate search query answers.
Java	soft	the main programming language used to program TXM.
keyword	com	the central column of a concordance that display all the occurrences of the search query aligned vertically.
language	mod	the main natural language in which a text or a corpus is written.
lem	mod	See lemma.
lemma	mod	the dictionary entry of a word.
lemmatizer	soft	a software component giving the dictionary entry to every word of a text.
lexicon	com	the action of listing all the possible word forms, or other word properties, in a corpus and their frequency.
literal	exp	a character taken as it is in a search query.
loader	com	a software component implementing a process to import a corpus into the platform from its source.
localization	int	the interface of TXM can be read in different languages, determined by the localization preference.
match	met	an occurrence of a search query in a corpus.
metadata	mod	the properties of a whole text or document. Each metadata has a name, a type and a value.
modifier	exp	a special character used to express a different meaning of a

TXM Reference Manual 0.5

		search query (for example 'ignore caps').
Multext	nlp	a European standard morphosyntactic tagset.
NLP	soft	for <Natural Language Processing>, software processing human language information in texts
occurrence	met	the appearance of an event in a corpus, like a word occurrence.
operator	exp	a special character expressing a particular constraint in a pattern in the search query language.
page	mod	a segment of text rendering, usually corresponding to a reference paper edition.
part	mod	an element of a corpus partition.
partition	mod	a decomposition of a corpus in several parts. The sum of all the parts of a partition is always the whole corpus. A partition is used to analyze contrasts between parts (like between dates of speeches, authors of texts, sections of a text, etc.)
pos	mod	for <[p]art [o]f Speech>, the main grammatical information of a word.
preference	int	all TXM commands have default parameters affecting their behavior. Some of those parameters can be edited in the 'Preferences' panel.
property	mod	an information about a lexical unit or a structural unit
query	com	the expression, by characters, of a pattern of word sequences combined with a pattern of word properties.
reference	int	an information displayed at the beginning of concordance lines coming from unit properties.
score	met	a numerical value indicating a statistical tendency.
script	soft	a file containing the description of a sequence of TXM actions to execute.
search query	com	the expression, by characters, of a pattern of word sequences combined with a pattern of word properties.
selection	met	a list of sequences of words. The search engine returns a selection.
sentence	nlp	an orthographically delimited sequence of words, generally computed by tokenizers.
source	mod	the original representation of a corpus in a specific format, possibly in several files and directories. For

TXM Reference Manual 0.5

		exemple the format can be TXT (raw text), XML or TEI.
specificity	com	the action of listing the most specific word forms, or other word properties, for each part of a partition according to the specificity quantitative model.
status line	gui	TXM displays temporary comments on operations in a line at the bottom left of the interface.
structural unit	mod	an element of the logical structure of a text. In TXM, all structural units are organized hierarchically: every unit is imbricated in an upper unit - until the 'text' unit. The lower and smaller structural units are above the lexical units.
T	met	the total number of occurrences in a corpus
tag	mod	the representation of element limits and their properties in the XML format.
tag	nlp	the morphosyntactic property of words
tagger	soft	an independent software component able to tokenize, grammatically tag and possibly lemmatize texts from their sources.
tagset	mod	the set of all the possible values for the morphosyntactic property of words.
TEI	fnt	for <Text Encoding Initiative>, the standard way of encoding texts. See http://www . tei -c.org . The TEI format is expressed in XML.
text	mod	a possibly structured homogeneous sequence of words, possibly described by properties. A text can be described by its metadata.
textometrie	met	the general methodology underlying TXM. See http://textometrie.ens-lyon.fr .
tokenizer	soft	a software component to compute word boundaries by their character properties, in source files.
TreeTagger	soft	an academic tagger.
TXT	fnt	the data format of raw text files (without annotations).
unit	mod	a leaf unit or lexical unit, or a structural unit of a text.
V	met	the total number of different graphical forms of a corpus.
vocabulary	com	the action of processing a lexicon or an index.
Weblex	soft	an academic software of textometry.
window manager	int	a software component helping to organize the interface windows.
word	mod	a lexical unit identified by its graphical form and its position in

TXM Reference Manual 0.5

		word sequences, generally computed by tokenizers.
workspace	int	the set of all the objects available to the user in TXM (corpus, sub-corpus...).
XML	fmt	the main data format for corpus source.

10 Bibliography

- Barclay, Kenneth A., et W. J. Savage. *Groovy programming: an introduction for Java developers*. Morgan Kaufmann Publishers, 2007.
- Benzécri, Jean-Paul, et al. *L'analyse des correspondances*. Paris: Dunod, 1973.
- König, Dierk, Andrew Glover, Paul King, Guillaume Laforge, et al. *Groovy in action*. Greenwich: Manning, 2007.
- Lafon, P. “Sur la variabilité de la fréquence des formes dans un corpus.” *Mots*, no. 1 (1980): 127-165.
- Venkat, Subramaniam. *Programming Groovy: dynamic productivity for the Java developer*. Pragmatic Bookshelf. Raleigh: Daniel H. Steinberg ed., 2008.

11 Index

Illustrations Index

Illustration 1: The general interface of TXM.....	13
Illustration 2: The Objects zone.....	14
Illustration 3: The Corpus view.....	14
Illustration 4: The File view.....	16
Illustration 5: The Toolbar.....	17
Illustration 6: The File menu.....	17
Illustration 7: The "Corpus" menu with, on the left, the corpus commands and, on the right, the partitions commands.....	18
Illustration 8: The "Tools" menu, for the corpus and the partition objects.....	18
Illustration 9: The Corpus Contextual Menu.....	19
Illustration 10: The results.....	22
Illustration 11: The Messages zone.....	23
Illustration 12: Import window.....	27
Illustration 13: DISCOURS Description.....	31
Illustration 14: DISCOURS Edition.....	33
Illustration 15: Navigation window between the parts editions.....	33
Illustration 16: Simple sub-corpus selection : build the sub-corpus of all the speeches of the De Gaulle president.....	34
Illustration 17: Assisted sub-corpus selection : build a sub-corpus of the texts of the 12th century in verse.	35
Illustration 18: Advanced sub-corpus selection : build the sub-corpus of all the speeches of the Pompidou president made in 1970.....	36
Illustration 19: Simple partition building : build a partition on every date of speech.....	37
Illustration 20: Building a partition on the DISCOURS corpus with the text date values.....	38
Illustration 21: Build a partition on every president for the year 1970.....	39
Illustration 22: Concordance Initial Search Form.....	40
Illustration 23: Building a query for the word "je" followed by a verb.....	41
Illustration 24: Concordance of the "je" word followed by a verb in the DISCOURS corpus.....	42
Illustration 25: Reference Pattern Dialog Box.....	44
Illustration 26: Cooccurents of the words beginning by "j".....	45
Illustration 27: Lexicon dialog box.....	46
Illustration 28: word forms frequency list of the DISCOURS corpus sorted alphabetically....	47
Illustration 29: Index initial dialog box.....	48
Illustration 30: Index word properties editor.....	48
Illustration 31: Index of the combination of the 'form' then 'pos' word properties for all the occurrences of the "pouvoir" lemma in the DISCOURS corpus.....	49
Illustration 32: Specificity for a partition dialog box.....	50
Illustration 33: Specificity of "j.*" word forms in the discourse type partition of the DISCOURS corpus.....	51

TXM Reference Manual 0.5

Illustration 34: Specificity graphic of the "je", "jeune"... word forms between discourse genres in the DISCOURS corpus.....	52
Illustration 35: Specificity scores of the word forms of the "Allocution radiotélévisée" discourse genre in the DISCOURS corpus.....	53
Illustration 36: Progression processing parameters for the "France" and "Algérie" words, in the DISCOURS corpus.....	54
Illustration 37: Progression graphic on the "France" and "Algérie" words in the DISCOURS corpus.....	55
Illustration 38: Graphics obtains from a lexical table, with the "Date" property, on the DISCOURS corpus.....	56
Illustration 39: Lexical table property selection.....	57
Illustration 40: Lexical table on the "Date" partition of the DISCOURS corpus.....	58
Illustration 41: Columns selection window.....	59
Illustration 42: TXM settings window.....	60

Index

Clipboard.....	11, 21
Command...10, 11, 13, 14, 15, 16, 17, 18, 19, 22, 23, 24, 25, 26, 28, 29, 31, 32, 33, 34, 36, 37, 39, 44, 45, 46, 47, 50, 51, 53, 54, 55, 57, 60, 62, 65, 71	
Concordance.....	11, 15, 21, 32, 39, 41, 42, 43, 44, 49, 50, 60, 62, 65
Context.....	42, 43, 45, 60
Contextual menu.....	15, 17, 19, 29, 42, 43, 44, 50, 52, 59, 65
Cooccurrence.....	15, 21, 44, 60
Cooccurrent.....	11, 21, 44, 45, 60
Corpus...6, 11, 13, 14, 15, 18, 21, 22, 23, 24, 25, 26, 29, 30, 31, 32, 34, 36, 37, 39, 40, 41, 44, 45, 46, 47, 50, 51, 53, 54, 62, 63, 64, 65	
Correspondence analysis.....	11, 15, 21, 37, 56
CQP.....	21, 36, 39, 40, 44, 45, 46, 47, 49, 51, 54, 62, 63, 64, 65
Description.....	18, 21, 57, 60, 64
Directory.....	7, 9, 10, 16, 21, 24, 25, 26, 27, 65
Document.....	6, 7, 26, 27, 28, 29
Explorer.....	13, 14, 17, 33
Export.....	17, 21, 25, 27, 29, 46, 47, 55, 59, 65
File.....	7, 8, 9, 10, 14, 16, 17, 21, 24, 25, 26, 27, 28, 29, 65
Flyover.....	28, 32
Folder.....	16
Format.....	11, 15, 24, 25, 26, 27, 28, 29, 44, 47, 60, 71
Graphic.....	52, 53, 54, 55, 56, 60
Groovy.....	16, 17, 26, 65, 66
HTML.....	11, 28, 32
Import.....	6, 11, 14, 21, 24, 25, 26, 27, 28, 29, 62, 63
Index.....	11, 15, 21, 26, 28, 46, 47, 49, 50, 57, 62
Keyword.....	11, 32, 40, 42, 43
Lemma.....	25, 27, 30, 46, 50, 55
Lemmatizer.....	27
Lexical pattern.....	11
Lexical table.....	11, 15, 21, 55, 57, 58
Lexicon.....	15, 21, 46, 53, 55
Loader.....	21, 24, 25, 26, 27, 28
Match.....	11, 41, 42, 51, 63, 64
Metadata.....	26, 27, 32
NLP.....	26, 29, 60, 71
Occurrence.....	41, 42, 43, 45, 64
Partition.....	11, 15, 18, 21, 33, 36, 37, 39, 47, 50, 51, 55, 57, 58
Pattern.....	21, 41, 42, 43, 54
Progression.....	21, 54
Property 11, 15, 21, 25, 26, 27, 28, 29, 30, 31, 32, 34, 36, 37, 40, 41, 42, 43, 44, 45, 46, 48, 50, 51, 52, 53, 54, 55, 56, 57, 60, 63	
Query.....	7, 21, 36, 39, 40, 41, 42, 43, 44, 45, 46, 47, 49, 50, 54, 62, 63, 64, 71
Raw.....	21, 24, 25, 26

TXM Reference Manual 0.5

Script.....	13, 16, 17, 26, 65, 66
Shortcut.....	22, 56
Shortcut,.....	17
Software.....	6, 7, 8, 11, 27, 29, 60
Specificity.....	11, 15, 21, 37, 50, 51, 53, 57
Tab.....	14, 23, 34, 36, 39, 56
Tag.....	11, 25, 29, 30, 46
Tagger.....	26, 27, 29, 41, 60, 63
Tagset.....	29, 30, 32, 41, 63, 64
TEI.....	24, 25, 29
Text.....	11, 14, 16, 17, 21, 24, 25, 28, 29, 30, 32, 33, 50, 65
Textométrie.....	6
Toolbar.....	17, 29, 44, 55
TXT.....	16
Unit.....	11, 25, 26, 27, 29, 30, 31, 34, 36, 37, 39, 40, 41, 44, 50, 51, 54, 57, 64
Vocabulary.....	11
XML.....	16, 24, 25, 26, 29