TeXstudio Development

1 Rules

This chapter describes some development rules.

1.1 Basic C++

The identifier should be named in these ways:

Style	Explanation
ClassSomething	First character and every word in uppercase for a *class*
identifierSomething	camelCase for all local variables and *members* (notice: no m_ prefix for texmaker compatibility)
headersomething.h	All header in lowercase
headersomething.cpp	sources, too
HEADERSOMETHING_H	include guards in uppercase
SOME_CONSTVALUE	const/enum values in uppercase ?????? (with prefix??? without???)

Order the include directives from high level to low level, e.g.:

```
#include "texmaker.h"
#include <qt>
```

instead of the other way around.

Insert only things in texmaker.cpp or smallUsefulFunction.h if it is really necessary. It is better to create a new .cpp/.h if you can't find one to put it in.

1.2 Compatibility

Don't use QT 4.4 or 4.5, only functions of Qt4.3.
If they are really necessary you could wrap them in #if, like:
`#if QT_VERSION >= 0x040500` für Qt 4.5.0
`#if QT_VERSION >= 0x040400` für Qt 4.4.0

But this prevents linking with older qts, so it is better to call hasAtLeastQt(4,5) of smallAndUseful, to check and then call the methods dynamically.
Use QMetaObject::invokeMethod for Slots/Signals and QObject::setProperty for properties

Some can be easily replaced, these include:

Example	Use Instead	Remark
QFormLayout	QGridLayout	
Q(String)List.append(Q(String)List)	<<	This doesn't apply to .append(T)
Q(String)List.length	QList.count()	
Q(String)List.removeOne	`if (i=QList.indexOf()>0) QList.removeAt(i)`	
QRegExp.cap const	QRegExp.cap	cap has first been made const in 4.5

Those don't exists in old qt, but should be used for the newer version, so they must be included in #if. (min version is the minimum version where it can be used)

Example	Min. version	Possible Replacement
QPalette::ToolTipBase	4.4	QPalette::AlternateBase
QPalette::ToolTipText	4.4	QPalette::Text

2 TODO

2.0.1 necessary before 2.2

TM changes

- interface disabled during compiling (says sources, i didn't see a effect)
- file chooser in status bar (TXS already has this in the toolbar, very non intuitive to activate) OBSOLETE: available as context menu on editor tabs

2.0.2 later

Bugs

- false syntax check error in `\abc123`
- `_` in movie command marked as math error
- latex checking says "cols in tab missing", although this does not even cause a warning by Latex. (perhaps there should be another marking overlay, for bad style, and this marked as that)
- Autocompletion of a reference name `\ref{foo` eats the next word
- In a trivial beamer document frame + env. completion only adds \begin, not \end
- completion window appeared at the wrong position (cursor almost at the right document border, window appeared top left)
- zooming in editor doesn't correctly synchronizes among all editors (because only one font is allowed in the application)
- unnecessary slowness: (possible duplicate setHeight call after adjustWidth)
- codesnippt mit placeholder einfügen und rückgängig machen plaziert cursor an falscher column position =¿ schreiben zerstört rückgängig (see failing, commented tests in codesnippet_t)
- env highlighting doesn't work if there are points in the env. name (e.g. `\begin{a..s}`)
- syntax checking highlighting bug (fixed?): error background not updated unless the cursor moves there (on Windows)
- bracket renaming doesn't work for brackets without content (like `[]`)
- stop bracket renaming if space is pressed
- revert order of brackets
- auto replace is not atomic for menu replace (e.g. select text, press shift+alt+f, multiple undos necessary to return to selection)
- Home/Ende keys don't work (siehe mrunix forum)
- adding new bib tex entries doesn't always update completer list;; also changing citations in latex doc also require bibtex call and aren't detectedd
- completer doesn't contain citation labels (this means if you have `\cite{abcd|` with — marking cursor position it doesn't suggest abcdef as citation even if `\cite{abcd}` is in the list)
- texworks pdf viewer changes it cursor when shift/alt is pressed, but that only works every 2nd time
- presentation mode doesn't hide cursor
- zooming in/out in continuous mode changes the current page
- switching between forward/backward search in the pdf previewer, sometimes doesn't find matches or finds them twice (e.g. in this development.pdf). Possibly a poppler bug

- synctex jumps to the wrong beamer page if only-commands are used
- hint by `\ref` parses html code. E.g. if you have

  ```
  <html><body>asas<img src=":/images/splash.png"><b>AAA
  \label{key}
  </b></body></html>
  ```

  in the latex file (if there is a security bug in the qt html renderer, it could become a security bug in TXS)
- qce crashes if the searchpanel is not deleted before the qcodeedit object is deleted (because the search as child of the editor is then automatically deleted after the editor and crashes when it calls the editor to clear it match overlays)
- Auto completion is sometimes unexpected (see unittests)

others

- QCodeEdit:
  - show "ignore all" for deleted files (for current session)
  - whitespace and symbol formats
  - option to change color of linenumber/find panels
  - Generate custom highlighting from the used custom completion cwl files. (important for syntax check, if you have a custom math environment you currently have to define it twice, once as custom highlighting(to syntax check multi line environments) and in the cwl (to check single line envs))
  - DIFF: jump to next mark, warn if file is saved while diff overlays exist
  - Deleting placeholders by pressing repeadetly backspace after one
  - save zoom options
  - if replace all is restarted from scope begin, show the count of replaced occurrences before the scope restart and afterwards.
  - show tooltips when scrolling (line, section, chapter)
  - RTL support like in texmaker 1.9.0/1 (but there it was removed in 1.9.2) (import fy to unicode)
  - Auto Closing of `$` (problem: how to detect if there follows a closing `$`)
- Latex Parser:
  - move everything in the `Latexparser/reader` class
  - NextWord with cookie state (to recognize multi line verbatim)?? Or read from qce?
  - Understand some TeX Things (e.g. `^^5c^^5c` equals `\\` if not in verbatim)
- better structure view:
  - select section title when double clicked of one?
  - open whole structure when loading a (single) file
  - parsing not loaded files
  - show section/figure/table numbers
  - should cache old parsing, don't reparse unmodified files
  - could use .aux files
  - should not execute several regex after each other on the same line (custom parser? mixed regex?)
- Synonym dictionary:
  - remember window size
  - faster contains checks using Tries
  - support for online lookup (e.g german: wortschatz.uni-leipzig.de)
- Labels/References:

  - show all refs using a label and allow easy jumping to them (e.g selection window? or ctrl+click on ref jumps to label; ctrl+alt+f does it already, perhaps add a mouse event to action event mapper)
  - using a SoundEx suggestion list for wrong references/labels like in the spell checker
  - a reference/label check dialog like the old spell checking dialog

- Citations:
  - unit tests for bibtex parser
  - show SoundEx suggestion for wrong citatinos
  - check environment paths
    (e.g. `BIBINPUTS = C:\My Documents\CDMA\Papers\Bib;C:\My Documents\GPS\Papers\Bib`)

- Better new document wizard (look at kile)

- Completion
  - file name completion for bib tex files
  - if completing a filename, and the selected file has the same extension as the current filename, remove the existing extension, show preview image
  - auto close last environment/bracket
  - complete command options after command (files for includegraphics, environsments for begin, citations, labels, packages, tikzpictures)
  - Word completion over all open documents
  - Show definition of `\newcommand` commands in the completion list as hint
  - `\begin{Umgebungsname}` should not add `\end{Umgebungsname}` if that \end is already in the next/a later line
  - Open completion list of xyz if cursor is after `\xyz{` (= ignore `{}` before cursor)
  - treat \abc{%¡xyz%¿} and \abc{%¡def%¿} as identical and ignore one of them (in the list of possible completions)
  - Understand `newcounter, tex: \newcount \newfont \def, \edef, \xdef, \gdef`
  - Detect completion within tikz/pstricks/... environments and show a list of corresponding tikz/... commands
  - text completion should show parameters of commands (e.g. reference names) even if they aren't currently used
  - merging text completion and spell checker cache?

- brackets:
  - Menu command to complete current/last environment/bracket
  - inverting parenthesis selection should support multiple begin/end pairs
  - find missing bracket command (instead of just misplaced brackets)

- GUI:
  - shortcut for always ignore word (in context menu)
  - grammar suggestions should show reason like the mouse over (with tooltip? statusbar? smaller font?)
  - differenciate between always-ignore-this-word-here and always-ignore-this-word-everywhere (add to dictionary)
  - search for latex commands in the completion package list; list for every available package the contained commands
  - new tm 1.9 output view (icons left instead of tabs + next/previous error buttons)
  - Customizable symbol lists
  - combo boxes (dropdown tool buttons) instead of several icons for each command (like 1.9 or kile)
  - toolbar icon size should be customizable

- Preview:

  - allow real zoom (regenerating png)
  - show errors

- Internal PDF-Previewer:
  - show pdf title and full path in windows title bar
  - three grid next page modes: single page step, grid row, full grid (or allow arbitrary steps? renderer already supports them)
  - presentation mode could have a "¿jump to page"¡ feature (for questions after a presentation); which is synchronized over all views (s.t. you can see different things on the computer screen vs. beamer, but still don't have to move the mouse cursors from beamer to screen to select page)
  - alternative: tool where left/right clicks goes to the next/prev page (i.e. a tool default in presentation mode, but usable in all views)
  - (Okular) presentation tools: mouse cursor as laser pointer, drawing on presentation (in tool menu at the top edge), page number display when switching
  - mark source bookmarks in pdf preview
  - mark contents (section pages) in presentation clock as lines
  - It should be possible to set a filename in the call command and a destination synctex-line
  - Shortcuts of pdf previewer should be customizable (managedMenu)
  - Toolbars should also be customizable
  - There should be an continuous display mode (scrollbar.size = pageheight*pagecount, and currentpage = scrollbar.pos/pageheight).
  - options menu entry of the pdf previewer should open the config dialog on a preview page
  - ctrl+click on a word should jump to the exact word in the source instead of an approximate position and it should work with formulas (is this even possible?)

- Checker:
  - Way to disable repetition checker (only in math. envs., globally, for a certain word)
  - Text analyse should have a list of environments to ignore and if it should read math/envs
  - Silbentrennung: `http://homepage.ruhr-uni-bochum.de/Georg.Verweyen/silbentrennung.html`
  - Respect `\verb` or even multiline `\begin{verbatim}`
  - LaTeX Checker (calling lacheck?)

- Log files:
  - if there are several errors, should it jump to the first or the one after the current cursor position? If the errors are in different files, should it jump to another file? which one? Should it jump to warnings? (current in r500 it only jumps to the next error in the same file)
  - More log: What do we do with errors in other files when we are not in master mode?
  - Is it a good idea to change the tab from "¿log file"¡ to "¿messages"¡ if another tex file is opened and we are not in master mode?
  - don't jump away from log file (what did that mean???????)

- Building
  - show stdout for asymptote
  - Is the output of (pdf)latex shown? should it? Show if it crashes (waiting for pipe, monitor cpu usage)
  - Improved recognizing of tool paths, checking of correct tool settings, custom build actions: planned (although former is finished for miktex+ghostscript)
  - better browser detection in WebPublish dialog
  - . command tooltips (right section?)

- Scripting
  - access to undo/redo stack (for debugging purposes)

  - qdocumentsearch access (is it done??)

- Plugins
- Recognizing hand written symbols (online lookup in detextify or Windows 7 mathpanel)
- New Tabular Dialog (of kile)
- Having functions to select environments/blocks/sections, changing `$$` to `\[\]`, or to `\begin{something}` (waiting for qce2.3 syntax engine)
- single instance: open the documents in the order the second instances were called instead in the order theirs messages arrive in the first instance (using a queue like described in 2877037)
- Unit Tests (gui, qeditor, qdocument,..)