



VYM

—

View Your Mind

©Uwe Drechsel

April 24, 2006

Contents

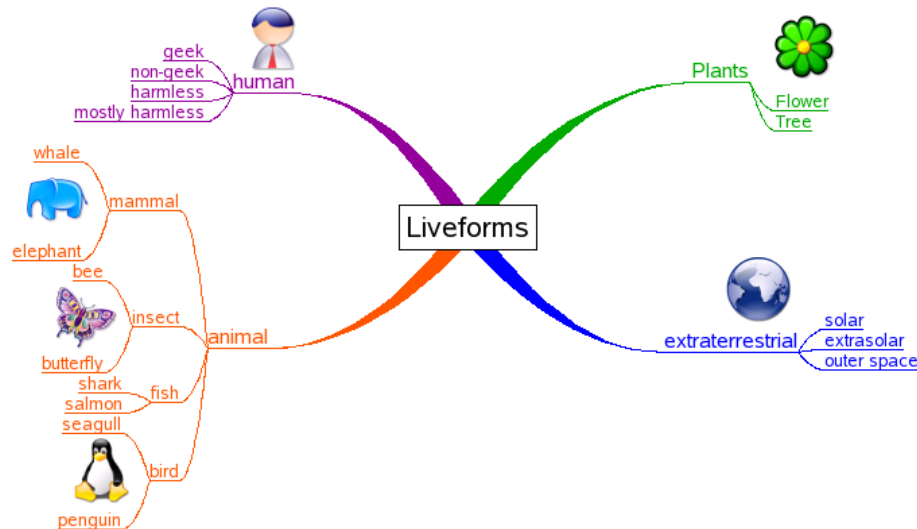
1	Introduction	4
1.1	What is a VYM map?	4
1.2	Why should I use maps? Time, Space and your Brain.	4
1.3	Where could I use a map?	5
1.4	What you shouldn't do with a map...	5
1.5	Internet Ressources	5
2	Concept of VYM	5
2.1	Windows: mapeditor and noteeditor	5
2.2	Menus and Context menus	6
2.3	Toolbars	6
2.4	Maps	6
3	Mapeditor	7
3.1	Start a new map	7
3.2	Navigate through a map	7
3.3	Modify and move branches	9
3.4	The right side of your brain - colors and images	10
3.5	Background design	11
3.6	Links to other documents and webpages	12
3.7	Multiple maps	13
4	Noteeditor	13
4.1	States	13
4.2	Import and export notes	13
4.3	Edit and print note	13
4.4	RichText: Colors, paragraphs and formatted text	14
4.5	Fonts and how to quickly switch them	14
4.6	Find text	14
4.7	Paste text into note editor	14
4.8	Advanced actions	14
5	Hello world	15
5.1	Import	15
5.2	Export	15
6	Advanced Editing	17
6.1	Modifier Modes	17
6.2	Hide links of unselected objects	18
6.3	XLinks	18
6.4	Adding and removing branches	19
6.5	Adding a whole map or a part of a map	19
7	VYM on Mac OS X	19
7.1	Overview	19
7.2	Contextmenu and special keys	20
7.3	Viewing external links	20
A	Starting VYM	20
A.1	Path to ressources	20
A.2	Command line options	20

B	Contributing to VYM	20
B.1	Getting help	21
B.2	How to report bugs	21
B.3	Compiling from the sources	21
	B.3.1 Getting the sources	21
	B.3.2 The Qt toolkit	21
	B.3.3 Compiling VYM	22
B.4	VYM file format	22
B.5	New features	22
B.6	New languages support	22
B.7	New export/import filters	23

1 Introduction

1.1 What is a VYM map?

A VYM map (in short words *map*) is a tree like structure:



Such maps can be drawn by hand on a paper or flip chart and help to structure your thoughts. While a tree like structure like above can be drawn manually VYM offers much more features to work with such maps. VYM is not another drawing software, but a tool to store and modify information in an intuitive way. For example you can reorder parts of the map by pressing a key or add various information like a complete email by a simple mouse click.

Once you have finished collecting and organizing your ideas, you can easily generate for example a presentation in Open Office based on a map.

1.2 Why should I use maps? Time, Space and your Brain.

Space

A map can concentrate a very complex content on little space e.g. a piece of paper. It helps to use both sides of your brain: the logical side and also your creative side (e.g. by using pictures, colors and keywords in a map, so called *anchors*). It is a technique to organize the way you think: It can help you by developing, sorting and memorizing your thoughts.

Time

Because you just use keywords and drawings, it is much faster than good old fashioned notes. Your brain memorizes things by associating them to other things – a map makes use of those connections and stimulates new associations.

Your Brain

In 1960 the Prof. ROGER SPERRY discovered that both hemispheres of the human brain have different tasks (of course both of them basically *can* do the same):

Left side	Right side
<ul style="list-style-type: none"> • verbal speech and writing • numbers • logical thinking • analyzing and details • science • linear thinking • concept of time 	<ul style="list-style-type: none"> • body language • visual thinking, day dreams • intuition and emotion • overview of things • creativity • art, music, dancing • non-linear thinking, connecting things • spatial awareness

In our science oriented society we have learned to mainly rely on our left side of the brain, the "rational" one. In other cultures, especially like the native americans and other "old" cultures, the right side is much more important. Maps are just one way to stimulate the other side and make use of additional ressources we all have.

1.3 Where could I use a map?

Here are some examples, how you can use those maps

- to prepare articles, papers, books, talks, ...
- to sort complex data
- to memorize facts, persons, vocabulary, ...
- to sort emails, files and bookmarks on your computer
- to moderate conferences

1.4 What you shouldn't do with a map...

A map drawn by somebody shows the way the author thinks. There is no right or wrong in the way it is drawn, so there is no way to criticize it. "It is, what it is" (F. LEHMANN).

1.5 Internet Ressources

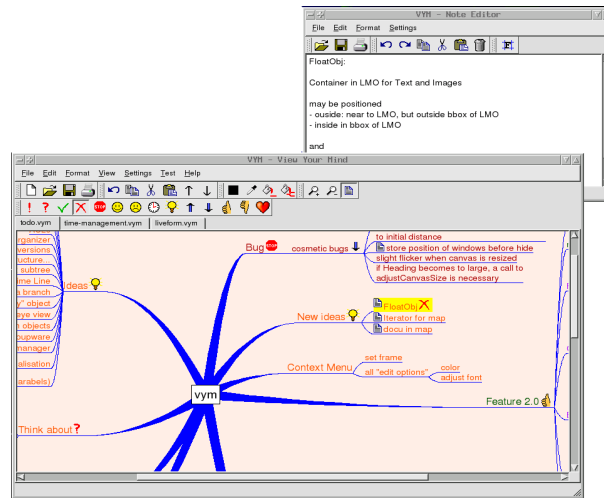
A good starting point to learn more about maps in general is Wikipedia:

- English: http://en.wikipedia.org/wiki/Mind_map
- German: <http://de.wikipedia.org/wiki/Mindmap>

2 Concept of VYM

2.1 Windows: mapeditor and noteeditor

VYM uses two windows: an editor for the map itself and another one for notes, which are part of the map. Let's call them *mapeditor* and *noteeditor*:



Usually you will work in the *mapeditor* by just adding new branches, moving around and reordering them. The various ways to do this will be explained in 3. You can store additional information e.g. the content of a email easily in a *branch*: Just type or copy&paste it into the *noteeditor*. Working with notes is explained in 4

2.2 Menus and Context menus

On top of each window you find the menubar. The options you find there are similar to those you are used from other applications. Note that many (and even more) options are available via *context menus*. Those are available if you right-click onto an object in a map (on Mac OS X Command-Click).

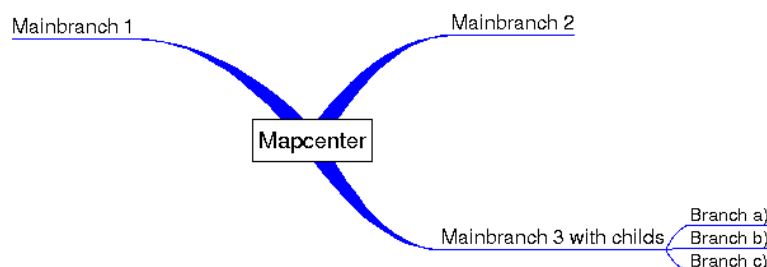
2.3 Toolbars

The toolbars in the mainwindows give quick access to many functions and also visualize the state of an object. For example a part of of the map can be hidden when the map is exported into an Open Office presentation. To show this the branch in the map will have a little cloud symbol, which is also "switched on" in the toolbar.

Note that you can reposition all toolbars by simply grabbing them. For example you can move the flags-toolbar from its original horizontal position on top of the mapeditor to a vertical position on the right side. You can even detach it and make it "float" separate from the other windows. Or just insert it again at its original position.

2.4 Maps

The map itself has always a *mapcenter*. The mapcenter has *branches* just like the trunk of a tree. Each branch in turn may have branches again.



We will call a branch directly connected to the mapcenter a *mainbranch*, because it determines the position of all its child branches.

The mapcenter and the branches all have a *heading*. This is the text you see in the mapeditor. Usually it should just be one or a few words, so that one can easily keep track of the whole map.

In the toolbar above the mapeditor you see various symbols.



These are called *flags* and can be used to mark branches in the map, e.g. if something is important or questionable. There are also more flags set by VYM automatically to show additional information, e.g. when a exists for a particular branch.

By default some of these flags are set exclusively e.g. when the "thumb-up" flag is set, then the "thumb down" is unset and vice versa. You can change this default behaviour in the settings menu.

3 Mapeditor

3.1 Start a new map

After VYM is started two windows will open: the mapeditor and the noteditor. Usually you will work in both windows, but at the moment we will just need the mapeditor.

Select the mapcenter "New map" in the middle of the mapeditor by left-clicking with the mouse. It will turn yellow to show that is selected. There are several ways to add a new branch to the center:

- Using the mouse: Open the context menu by clicking with the right mouse button (CTRL-Click on Mac) onto the mapcenter and choose Add → Add as child
- Press [Ins] or [A]

A new branch will appear and you will be able to type the heading of the branch. Finish adding the new branch by pressing [Enter]. Sometimes it comes handy to add a new branch above or below the current one. Use [Ins] together with [Shift] or [Ctrl]. It is also possible to add a branch in such a way, that the current selection becomes the child of the new branch, which is like inserting it *before* the selection. This can be done using the context menu.

3.2 Navigate through a map

Select branches

To select branches you can use the left button of your mouse or also the arrow keys. Depending on the *orientation* of a branch type [←] or [→] to get nearer to the mapcenter or deeper down into the branches. Within a set of branches, let's call them a *subtree*, you can use [↑] and [↓] to go up and down. You can also use [Pos1] and [End] to select the first and last branch.

Zoom a map

While adding more and more branches the size of the map may become bigger than the mapeditor window. You can use the scrollbars on the right and the bottom of your mapeditor window to scroll, but it is easier to just scroll using the left

mouse button: Click onto the *canvas* itself, the empty space somewhere between the branches. The mouse pointer will change from an arrow to a hand, now move the visible part of the map to show the desired part.

If you select branches using the arrow keys, the map will scroll to ensure that the selected branch is always visible.

Working with huge maps, the *zoom*-function comes in handy: You can use

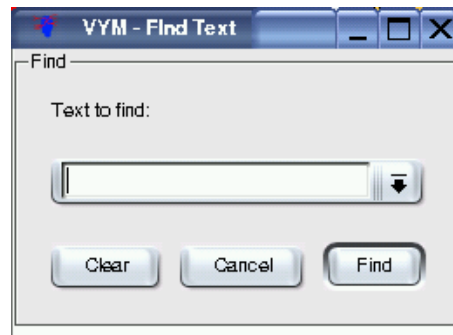
- from the menu View → Zoom
- the toolbar buttons



The crossed magnifying lens resets the zoomed view to its original size.

Find Function

With huge maps there is the need to have a find function. Choose Edit → Find to open the Find Window:



The text you enter here will be searched in all the headings and also in notes. Everytime you press the "Find"-button it will look for the next occurrence, which then will be selected automatically. If the search fails, there will appear a short message "Nothing found" or a few seconds in the *statusbar* on the bottom of the mapeditor.

Keep the overview – scroll a part of the map

A very big subtree of a map e.g. a branch with hundreds of childs makes it very hard to keep an overview over the whole map. You can hide all the childs of a branch by *scrolling* it – this function is also often called *folding*. Think of the whole subtree as painted onto a big newspaper. You can scroll the paper to a small roll, leaving just the headline readable.

To scroll or unscroll a branch and its childs, press the

- [Scroll] key or [S]
- press the middle-mouse button or
- choose the little scroll from the toolbar.

If you select parts of a scrolled branch e.g. using the find function or by using the arrow-keys, it will unscroll temporary. This is shown as a scroll with a little hour glass. If the temporary unscrolled part is not longer needed, it will be hidden again

automatically. It is also possible to unscroll all branches using "Edit→Unscroll all scrolled branches".

You can also hide parts of the map while exporting it e.g. to a webpage or a presentation, see 5.2 for details.

3.3 Modify and move branches

Modify the heading

You can edit the heading by selecting the branch and then

- pressing [Enter]
- double-clicking with left mouse.

Just type the new heading (or edit the old one) and press [Enter].

Move a branch

The easiest way to move a branch is to select it with left-mouse and drag it to the destination while keeping the mouse button pressed. Depending on the branch it will be

- moved to the destination or
- *linked* to a new *parent* (mapcenter or branch)

If you drag the branch over another one or over the mapcenter, you will notice that the link connecting it to the old parent will be changed to lead to the new parent which is now under your mousepointer. If you release the button now, the branch will be relinked.

If you release the button in the middle of nowhere, the result will depend on the type of branch you are releasing:

- A mainbranch is directly connected to the mapcenter. It will stay on its new position.
- An ordinary branch will "jump" back to its original position.

Thus you can easily rearrange the layout of the mainbranches to avoid overlapping of their subtrees. There is another convenient way to move branches, especially if you want to *reorder* a subtree: You can move a branch up or down in a subtree by

- pressing [↑] and [↓]
- selecting Edit → Move branch
- clicking on the toolbar buttons:



There is yet another way to move branches: If you press [Shift] or [Ctrl] while moving with the mouse, the branch will be added above or below the one the mouse pointer is over. This helps also to reorder a map.

3.4 The right side of your brain - colors and images

Change color of a heading

You can also use colors to put more information into a map, e.g. use red, green and more colors to prioritize tasks. Again you can

- use the menu and choose e.g Format 'Set Color
- use the toolbar

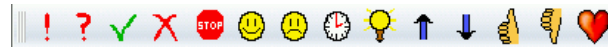


The first button (black in the graphic above) shows the actual color. Clicking on it let's you choose another color. You can also "pick" another color by selecting a branch with the desired color and using the "pick color" button. Both of the buttons showing a bucket actually put the current color to the selected branch. While the first one just colors the heading of the selection, the last one also colors all the childs of the selected branch.

A very useful function is the "copy color" using the mouse: Select the branch which should get the new color, then press [Ctrl] and simultaneously click with left-mouse on another branch to copy its color to the first one. Here the childs of the selection also will get the new color, if you just want to color the selection itself, additionally press [Shift].

Use flags

VYM provides various flags. You see them in the toolbar on top of the mapeditor window. (Note: Like all toolbars you can also move them to the left or the right side of the window or even detach them. Just grab the very left "dotted" part of the toolbar with your left-mouse button.)



If you have a branch selected, you can set any number of flags by clicking them in the toolbar. The toolbar buttons change their state and always reflect the flags set in the selected branch.

Presently VYM uses two kinds of flags: *System Flags* and *Standard Flags*. The standard flags are those shown in the toolbar. System flags are set by VYM to indicate e.g. that there is additional information in a note (more on this in 4). Later versions of VYM may have another kind of flags, which may be edited by the user.

Images

The easiest way to add an image to a branch is by dragging it e.g. from a webbrowser to the mapeditor while a branch is selected there.

You can also add a image to a branch by opening the context menu of the branch choose "Add Image". A dialog window lets you choose the image to load. ¹ While an image is selected in the dialog, you can see a preview of the image. It is also possible to select multiple images.

¹Supported image types are: PNG, BMP, XBM, XPM and PNM. It may also support JPEG, MNG and GIF, if specially configured during compilation (as done when VYM is part of SUSE LINUX).

You can position the image anywhere you want, just drag it with left mouse. To relink it to another branch, press [Shift] while moving it. To delete it, press [Del].

If you right-click onto an image, a context menu will open which let's you first choose one of several image formats. Then a file dialog opens to save the image. Hint: This is used to "export" the image, it will be saved anyway in the map itself! You can also cut and copy images, but it is not possible to add objects to an image²

The option '**Use for export**' controls the output of exports e.g. to HTML: If set to no, the image won't appear in the *text* part of the output. This is useful for large images or if images are used as a kind of frame e.g. the famous cloud symbol around a part of the map. Those shouldn't appear in the middle of the text.

At the moment image support is preliminary: Images will be saved together with all the other data of a map in the `.vym`-file. Later versions will include more functionality like resizing the images, changing its z-value (put it into background) etc.

Frames

A frame can be added to a branch by clicking with the right-mouse button. A context menu will open, where you can choose the frame. At the moment just a rectangle resp. "No Frame" will be offered, nevertheless you can use images as frames. Have a look at the demo map `todo.vym` as an example, where the mapcenter is a cloud. You can use an external drawing program like `gimp` to create an image, preferable with an transparency channel, so that you can design frames which don't use a rectangular borderline, just like the cloud.

3.5 Background design

The design of the background of a map and also of the links connecting various parts of the map can be changed by

- Selecting Format from the menu
- Right clicking onto the canvas, which will open a context menu

Background color

The color is set (and also displayed) as "Set background color".

Link color

Links connecting branches can be colored in one of the following ways:

- use the color of the heading of the branch the links is
- use *one* color for all links. The default color is blue. leading to.

The latter can be set with "Set link color". Check or uncheck the "Use color of heading for link" option to choose one of the two designs for your map.

Link style

VYM offers four different styles for the appearances of links:

- Line

² Images are regarded as "extra feature". It would make working with the map much more complex if e.g. images could be linked to images.

- Parabel
- Thick Line
- Thick Parabel

The "thick" styles only draw links starting at mapcenter thick, the rest of the map is always painted "thin".

3.6 Links to other documents and webpages

VYM supports two kind of external links:

- Document, which will be opened in an external webbrowser
- VYM map, which will be opened in VYM itself

In addition to the external links there also internal ones, leading from one branch in a map to another one. Those are called *XLinks* and are explained in section 6.3.

Webbrowser

Modern Webrowsers like **konqueror** are able to display various types of files, both local or in the internet. To enter the URL of any document, right-click onto a branch or use the Edit Menu and choose "Edit URL". Enter the path to your document (or copy and paste it from your browser). Examples for valid paths are:

```
http://www.insilmaril.de/vym/index.html
file:/usr/share/doc/packages/vym/doc/vym.pdf
```

If an URL was entered, a little globe will appear in the branch. By clicking on the globe in the toolbar or the context menu an external browser³ will be started.



If you use konqueror as browser, VYM will remember the konqueror which was opened first by VYM. You can also press [Ctrl] and click to open the link in a new tab then.

If you want to keep bookmarks in a map, select a branch where you want to add the bookmark, then simply drag the URL from your browser to the map. Also you could use an existing heading as URL: Right click onto the branch and select "Use heading for URL".

VYM map

To link to to another map right click on a branch or choose "Edit → Enter VYM link". A file dialog opens where you can choose the map. A branch with a link is marked with



Clicking this flag in the toolbar or in the context menu of a branch will open the map in another tab (see 3.7 for working with multiple maps). To delete an existing link, just press the "Cancel" button.

Technical note: Internally VYM uses absolute paths, to avoid opening several tabs containing the same map. When a map is saved, this path is converted to a relative one (e.g. `/home/user/vym.map` might become `./vym.map`. This makes it fairly easy to use multiple maps on different computers or export them to HTML in future.

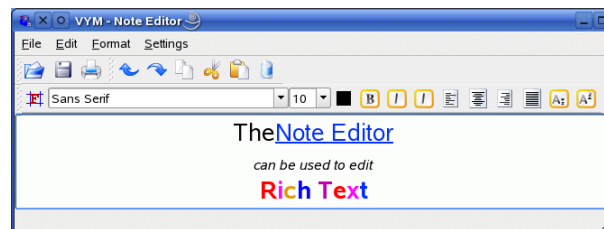
³ The browser can be changed in the Settings Menu.

3.7 Multiple maps

You can work on multiple maps at the same time. Each new map is opened in another *tab*. The available tabs are shown just above the mapeditor. You can use the normal cut/copy/paste functions to copy data from one map to another.

4 Noteeditor

If you want to save more text in a branch e.g. a complete email, a cooking recipe, or the whole source code of a software project, you can use the noteeditor.



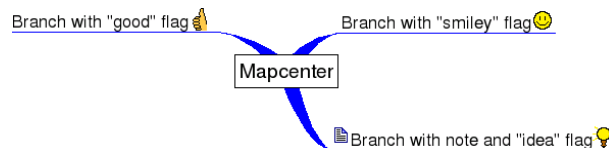
This editor displays text associated to a branch selected in the mapeditor. To visualize that there maybe is no text yet, the noteeditor shows different background colors depending on its state:

4.1 States

Before you can type or paste text into it, you have to select a branch in the mapeditor. Note that the background color of the noteeditor indicates its state:

- black: no branch selected
- grey: no text entered yet
- white: text is already available

To show you in the mapeditor itself that there is a note with more information for a particular branch, a little note flag will appear next to the heading of the branch. See the lower branch on the right side:



4.2 Import and export notes

The note is always saved automatically within the vym file itself. Nevertheless sometimes it is nice to import a note from an external file or write it. Use "File→ Import" and "File→ Export" to do so.

4.3 Edit and print note

Editing works like in any simple texteditor, including undo and redo functions. You can delete the complete note by clicking the trashcan. Only the note itself is printed by clicking the printer icon.

When pasting into the editor using the X11 copy&paste mechanism, the editor will create a paragraph for each new line. Usually this is not wanted, so there you

can convert all paragraphs into linebreaks by using Edit → Remove Paragraphs or [ALT-X].

4.4 RichText: Colors, paragraphs and formatted text

VYM supports formatted text (QT Rich Text) in the noteeditor since version 1.4.7. Colors and text attributes (e.g. italic, bold) can be set with the buttons above the text. The text itself is divided in paragraphs. For each paragraph the format can be set (e.g. centered, right). A paragraph is ended when a [Return] is entered. If you just want to begin a new line, press [CTRL-Return].

4.5 Fonts and how to quickly switch them

The noteeditor is ment to be used for simple notes, not really as full featured text editor. Because of many requests VYM supports now formatted text in the noteeditor⁴ Two default fonts are supported which can be set in the Settings menu. One is a fixed width font, the other has variable width. The fixed font is usually used for emails, source code etc. while the variable font is used for simple notes, where one doesn't need fixed character widths. Both fonts can easily switched using the following symbol from the toolbar:



In the Settings menu both fonts can be set and also which font should be used for default.

Additionally to the default fonts any font installed on your system can be used. Please note, that the chosen font also will be used for HTML exports, so you should only use fonts which are available generally.

4.6 Find text

The noteeditor itself has no Find function, use Find in the mapeditor, which will also search all notes (see 3.2).

4.7 Paste text into note editor

Often you will paste text into the editor from another application e.g. an email. Normally VYM will generate a new paragraph for each new line. This usually is not what you want, so you can choose from the menu

4.8 Advanced actions

Edit → Convert subsections:

This turns subsections in selected text (or all text, if nothing is selected) into linebreaks. This is especially useful for snippets of source code.

Edit → Join Lines:

Tries to format text, so that empty lines are used to delimit paragraphs. This is done for selected text (or all text, if nothing is selected). Especially useful for text like emails, meeting minutes etc.

⁴ VYM uses the QRichtText format, which is basically a subset of the formatting provided in HTML.

5 Hello world

This section is about how VYM can interact with other applications. Many applications meanwhile can read and write their data using XML, the eXtensible Markup Language. VYM also uses XML to save its maps, see [B.4](#) for a more detailed description.

So if your an application understands XML, chances are good that someone could write import/export filters for VYM . Volunteers are always welcome ;-)

5.1 Import

KDE Bookmarks

The integrated bookmark editor in KDE is somewhat limited, so why not use VYM to maintain the bookmark mess? To create a new map containing your current KDE bookmarks just choose

- File → Import → KDE Bookmarks

Mind Manager

VYM has currently a very basic import filter to convert maps created by *Mind Manager*⁵ into VYM maps. Notes and pictures are not converted at the moment. You can import files with

- File → Import → Mind Manager

Directory structure

VYM can read a directory structure. This is mainly for testing VYM e.g. to easily create huge maps used for benchmarks (yes, there is still room to optimize VYM ;-)

5.2 Export

Often you don't want to export the whole map, but just parts of it. For example you may have additional info you want to talk about in a presentation, while those parts should not be visible to the audience. To achieve this you can "hide" parts of the map during exports by setting the "hide in export" flag.



You can toggle this flag in the toolbar or by pressing [H]. Note that there is a global option in the settings menu to toggle the use of this flag. By default the flag is enabled.

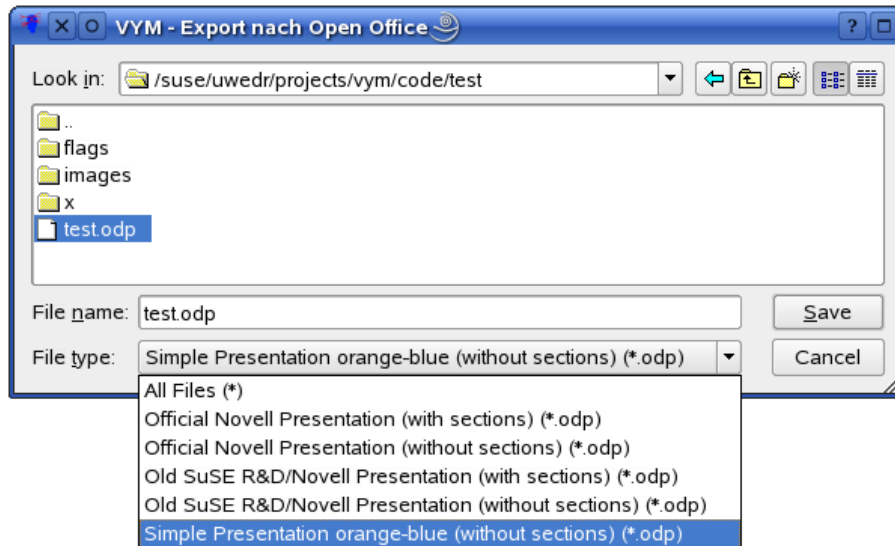
Open Office

Open Office beginning with version 2 uses the so called "Open Office Document Format", which can be written by VYM . The options are currently limited, but it is possible to export presentations which can be opened in Open Office Impress. By selecting

- File → Export → Open Office

you get a file dialogue where you can choose the output file and the file type:

⁵Mind Manager is a professional software by Mindjet. Both names are registered trademarks by Mindjet. For more information see their website at <http://mindjet.de>



The file types represent various templates, which can be created with some manual work from an existing Open Office document. The structure of VYM map is then inserted into a template. There are some limitations at the moment:

- VYM can't take care of page lengths, so you have to check and probably reedit in Open Office to avoid text running over the end of a page
- Images and flags are not used at the moment
- Notes are just written as plain text, without RichText

Some of the templates make use of *sections* e.g. insert the headings of mainbranches as chapters for sections into the presentation.

Image

VYM supports all image formats which are natively supported by the QT toolkit: BMP, JPEG, PBM, PGM, PNG, PPN, XPM, and XBM. For use in websites and for sending images by email PNG is a good recommendation regarding quality and size of the image. VYM uses QTs default options for compressing the images.

ASCII

Exporting an image as text is somewhat experimental at the moment. Later this will probably be done using stylesheets. So the output may change in future versions of VYM .

L^AT_EX

VYM can generate an input file for L^AT_EX. Currently this is considered as experimental, there are no options (yet). By selecting

- File → Export → L^AT_EX

you will be asked in a file dialog for the name of the output file. This file should be included in a L^AT_EX document using command

```
\include{inputfile.tex}
```


XHTML (Webpages)

This is the format you want to use to create a webpage. For an example have a look at the VYM homepage: www.InSilmaril.de/vym

Some explanation how this works: Before a map is exported as XHTML, it will be first written as XML into a directory (see 5.2). Then the external program `xsltproc`⁶ will be called to process the XML file and generate HTML code. A dialog allows to set various options:

- **Include image:** If set, VYM will create an image map at the top of the HTML output. Clicking on a branch in the map will jump to the corresponding section in the output.
- **Colored headings:** If set to yes, VYM will color the headings in the text part with the same colors like in the map.
- **Show Warnings:** If set to yes, VYM will ask before overwriting data.
- **Show output:** This is useful mainly for debugging. It will show how the processing of the XML file works by calling the external `xsltproc`.

Additionally the paths to the CSS and XSL stylesheets can be set. By default on SUSE Linux they will be in `/usr/share/vym/styles`.

XML

The map is written into a directory both as an image and as XML. The directory is set in a file dialog. If the directory is not empty, you will be questioned if you risk to overwrite its contents.

It is possible to export different maps into the same directory. Each file generated will have the map's name as prefix, e.g. `todo.vym` becomes `todo.xml`, `todo.png`, `todo-image-1.png` and so on. This is useful if e.g. for a website several combined maps have to be stored in the same directory.

Export a part of a map

Select a branch you want to export together with its childs, then open the context menu and choose *Save Selection*. This will create a file with the postfix `.vyp`, which is an abbreviation for 'vym part'.

6 Advanced Editing

VYM offers a few shortcuts to let you work more quickly and literally to connect any branches directly.

6.1 Modifier Modes

Modifiers are for example the [Shift]- or the [Alt]-keys. When pressed while doing actions with the mouse, they will cause VYM to use a "modified" action. E.g. you can move branches with mouse. If [Ctrl] or [Alt] is pressed while releasing the branch, it will be added above/below the target, not as child of the target.

Without a modifier pressed, the first click on a branch just selects it. For the behaviour of the [Ctrl] modifier there are several options, which can be set from the modifier toolbar:

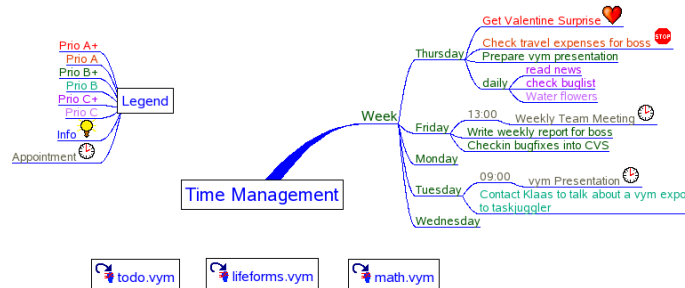
⁶On SUSE Linux `xsltproc` is installed by default.



The default is to copy the color from the clicked branch to the already selected branch. In the toolbar shown above the default modifier is selected, namely to copy the color of a branch. The second modifier let's you easily copy a whole branch with a single click. The third modifier lets you create *xLinks*, which will be explained in the next section.

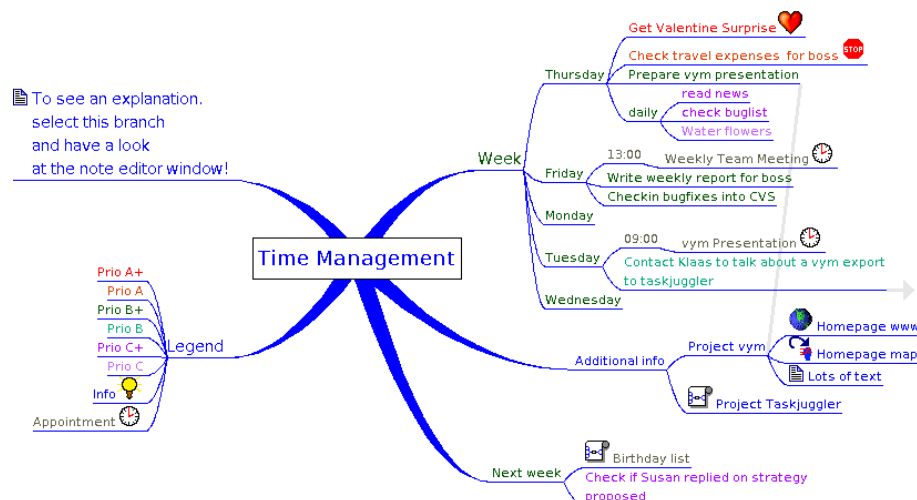
6.2 Hide links of unselected objects

Sometimes it would be useful to position a branch freely, just like a mainbranch or an image. Though this is not possible (yet) for all branches, you can use a mainbranch and hide its connecting link to the mapcenter. This can be used e.g. for legends or a collection of vymLinks pointing to other maps:



6.3 XLinks

So far all the data in the VYM map has been treelike. Using xLinks you can link one branch to any other, just like attaching a rope between two branches in a real tree. This is especially useful in complex maps, where you want to have crossreferences which don't fit on the same visible area, which fits on your screen. The following example, which is part of the VYM package, still fits on one screen, but shows how data can be crosslinked. In the graphics there is a link from a task (prepare a presentation) to general information:



Note that an xLink which points to a branch that is not visible (because it is scrolled), is just show as a little horizontal arrow. In the screenshot above have a look at the 'Tuesday' branch.

Create a xLink

Choose the link mode from the modifier toolbar (by clicking or pressing [L]). Select the branch, where the xLink should start. Press the modifier key [Ctrl] and simultaneously click on the branch where the link should end. (The link is already drawn before you release the mouse key). If you release the mouse over a branch the xLink becomes permanent.

Modify or delete a xLink

Open the context menu of a branch and select 'Edit xLink'. A submenu contains all the xLinks of the branch (if there are any). They are named like the branches, where they end. Choose one and the xLink dialogue opens, where you can set color, width and also delete the xLink.

Follow a xLink

In a complex VYM map it sometimes comes handy to jump to the other end of a xLink. You can do this by opening the context menu of the branch and clicking on 'Goto xLink' and selecting the xLink you want to follow.

6.4 Adding and removing branches

The context menu of a branch shows some more ways to add and delete data e.g. you can delete a branch while keeping its childs. The childs become linked to the parent of the previously removed branch. Similar branches can be inserted into existing maps. For keyboard shortcuts also have a look at the context menu.

6.5 Adding a whole map or a part of a map

Select a branch where you want to add a previously saved map (`.vym`) or a part of a map (`.vyp`), then open the context menu and choose *Add* \longrightarrow *Import*. For the import you can choose between *Import Add* and *Import Replace*: The imported data will be added after the selection resp. replace the selection.

7 VYM on Mac OS X

7.1 Overview

Basically there are two ways to run VYM on Macs:

QT Mac Edition:

VYM here provides the well known Mac look and feel. VYM is available as zipped Mac OS X application. It has been compiled and tested in Mac OS 10.3, but should also work on Tiger. It is using the Mac version of Trolltechs QT library.

X11

VYM can also be run using the Linux version, but then menus and handling will also be those of the Linux version e.g. The menu bar will look different.

7.2 Contextmenu and special keys

Most Macs unfortunately just have a single mouse button. In order to show the context menu which usually would be opened with the right mouse button, you can click while pressing the [kommand]-key.

Especially on Laptops some of the keys usually used on PC keyboards seem to be missing. The QT-Mac Edition of VYM has its own keyboard shortcuts. To find the shortcuts just have a look at all the menu entries, the shortcut is visible next to an entry. Toolbar buttons also may have shortcuts, just position the mouse pointer over a button and wait for the little help window to appear.

7.3 Viewing external links

VYM on Mac uses the system call `/usr/bin/open` to view links. Mac OS determines automatically if the link is a pdf or www page and opens the right browser.

A Starting VYM

A.1 Path to ressources

VYM will try to find its ressources (images, stylesheets, filters, etc.) in the following places:

1. Path given by the environment variable `VYMHOME`.
2. If called with the local option (see [A.2](#) below), VYM will look for its data in the current directory.
3. `/usr/share/vym`
4. `/usr/local/share/vym`

A.2 Command line options

VYM has the following options:

Option	Comment	Description
v	version	Show version of VYM
l	local	Use local paths to stylesheets, translations, icons, etc. instead of system paths. Useful for testing
h	help	Show help
q	quit	Quit immediatly after startup. Useful for benchmarks.

You can also give several filenames at the commandline to let VYM open several maps at once.

B Contributing to VYM

So far I'd say I have written 98% of the code on my own. No surprise, that VYM exactly fits my own needs. Nevertheless I would like to encourage all users of VYM to contribute. Maybe not only with feature requests, but also with code, new import/export filters, translations etc. In this appendix I'll try to show how easy it is to expand the things you can do already with VYM . I really look forward to hear from you!

B.1 Getting help

Frequently asked questions

Please refer to the FAQ available on the VYM website:

<http://www.InSilmaril.de/vym/faq.html>

Mailinglists

There are two mailinglists: `vym-forum` is the VYM users forum to discuss various questions, while `vym-devel` is intended for people interested in contributing to VYM. You can view the archives and subscribe at

https://sourceforge.net/mail/?group_id=127802

Contacting the author

Especially for support questions please try the mailinglists first. If everything else fails you can contact the Uwe Drechsel at

vym@InSilmaril.de

B.2 How to report bugs

Though Sourceforge has its own bugreporting system, I'd rather prefer if you contact me directly (see [B.1](#)) or even better: You can file a bugreport in Bugzilla, the bugtracking system of openSUSE:

http://en.opensuse.org/Submit_a_bug

I build VYM regularly for openSUSE, so you may report it against a recent version there, even if you use another Operating System. Please don't forget to tell

- the exact steps needed to reproduce the bug
- the version and build date of VYM (see the Help → About VYM)
- hardware and Operating System

B.3 Compiling from the sources

B.3.1 Getting the sources

You find the latest version of VYM at the project site:

<https://sourceforge.net/projects/vym/>

There you can check them out of the source repository (CVS):

```
cvs -d:pserver:anonymous@cvs.sf.net:/cvsroot/vym checkout code
```

B.3.2 The Qt toolkit

Qt is C++ toolkit for multiplatform GUI and application development. It provides single-source portability across MS Windows, Mac OS X, Linux and all major commercial Unix variants. Qt is also available for embedded devices. Qt is a Trolltech product. For more information see

www.trolltech.com/qt

B.3.3 Compiling VYM

Make sure you have installed your Qt environment properly, see the Qt documentation for details. You need to have the Qt command `qmake` in your PATH-environment, then run

```
qmake
make
make install
```

The last command `make install` needs root-permissions. Of course it may be omitted, if you just want to test VYM .

B.4 VYM file format

VYM maps usually have the postfix `".vym"` and represent a compressed archive of data. If you want to have a closer look into the data structure map called `"mapname.vym"`, just uncompress the map manually using

```
unzip mapname.vym
```

This will create directories named `images` and `flags` in your current directory and also the map itself, usually named `mapname.xml`. The XML structure of VYM is pretty self explaining, just have a look at `mapname.xml`.

This XML file can be loaded directly into VYM , it does not have to be compressed. If you want to compress all the data yourself, use

```
zip -r mapname.vym .
```

to compress all data in your current directory.

B.5 New features

There are lots of features which might find their way into VYM . Together with VYM you should have received a directory with several maps e.g. on SUSE LINUX this is

```
/usr/share/doc/packages/vym/demos
```

where you find the map `todo.vym`. It lists quite a lot of things to be done in future. If you have more ideas, contact the development team at `vym-devel@lists.sourceforge.net`.

B.6 New languages support

In order to add a new language to VYM you need the sources (see [B.3.1](#)) and an installation of Trolltechs QT. A part of QT are the development tools, from those tools especially the translation tool "Linguist" is needed.

In some Linux distributions the development tools are in an extra package, e.g. on SUSE LINUX you should have installed:

```
qt3-devel.rpm
qt3-devel-doc.rpm
qt3-devel-tools.rpm
qt3-man.rpm
```

If you don't have QT in your system, you can get it from <http://www.trolltech.com> Once you are able to compile vym yourself, you can translate the text in vym itself by performing the following steps:

- Let's assume now your encoding is "NEW" instead of for example "de" for german or "en" for english
- Copy the file `lang/vym.en.ts` to `lang/vym.NEW.ts` (The code itself contains the english version.)
- Add `lang/vym.NEW.ts` to the TRANSLATIONS section of `vym.pro`
- Run Linguist on `vym.NEW.ts` and do the translation
- Run `lrelease` to create `vym.NEW.qm`
- Do a `make install` to install the new vym and check your translation

If you feel brave, you can also translate the manual. It is written in LaTeX, you just have to change the file `tex/vym.tex`. (Linguist and QT are not needed, but it is useful to know how to work with LaTeX and esp. `pdflatex` to create the PDF.)

Please mail me every translation you have done. I can also give you a developer access to the project, if you want to provide translations regulary.

B.7 New export/import filters

VYM supports various kinds of filters. Data can be written directly, inserted into templates or it can be written as XML data and then processed by XSL transformations.

Most of the import/export functionality is available in the classes `ImportBase` and `ExportBase` and subclasses. All of them can be found in `imports.h` and `exports.h`.

Direct import/export

An example for a direct export is the XML export. This method touches the implementation of nearly every object of VYM, so whenever possible you should better use a XSL transformation instead.

If you still want to know how it is done, start looking at `MapEditor::saveToDir` in `mapeditor.cpp`.

Templates

Templates have been introduced to export to opendoc format used e.g. by Open Office. While I read the spec (> 500 pages) about the format⁷ I had the feeling that I did not want to write the export from scratch. It would be too complex to adapt the styles to your own wishes, e.g. the layout.

Instead I analyzed existing Open Office documents. I found out that there are lots of redundant bits of information in a standard presentation, for example each list item is contained in its own list. In the end I came up with the default presentation style, which still could be simplified, just in case you have free time...

The existing templates are still work in progress, before you spent too much time developing your own style, please contact me. Basically the following steps are needed to build your own style:

1. Create an example in Open Office. Use a title, authors name, page heading etc. which you can easily grep for in the output file.
2. Unzip the Open Office document into a directory.

⁷ <http://www.oasis-open.org/>

3. The main file is called `content.xml`. All data is in one single line. You can split the XML tags using the script `scripts/niceXML`, which is part of the VYM distribution.
4. Copy the output of `niceXML` to `content-template.xml`.
5. Looking closer you will find lots of unused definitions, for example of styles. You can delete or simply ignore them.
6. Try to find your title, authors name. VYM will replace the following strings while exporting:

```
<!-- INSERT TITLE -->    title of map
<!-- INSERT AUTHOR-->    author
<!-- INSERT COMMENT -->  comment
<!-- INSERT PAGES-->     content of map
```

The content itself is generated in a similar way by inserting lists into `page-template`. Here the following substitutions are made:

```
<!-- INSERT PAGE HEADING-->  heading of a page (mainbranch or child of
                                mainbranch, depending on the use of sections)
<!-- INSERT LIST -->         all childs of the branch above
```

Currently images are exported and notes just will appear as text without formatting and colors.

XSL Transformation

VYM uses XSL transformations while exporting (e.g. XHTML) and importing data (e.g. KDE bookmarks). There is a little code needed to provide the GUI, the rest is done using the `.xsl` stylesheet and calling the `xsltproc` processor, which is part of libxslt, the XSLT C library for GNOME.