## XVim Overview

3. Create command from the event and operete on Vim

interface

4. Operete the command in the application dependent way

XVim Application Independent Layer (Vim abstracted structures are here)

Vim Operation Interface

XVim Application Dependent Layer (Currently this is Xcode 4 dependent layer)



1

Xcode (Or some other applications in the future\*)



2. Pass the event

Abstraction Layer defines what kind of event to be passed.

- I. Notify Events
  - Initialization
  - Key event
- Mouse event ... and so on

(This is currently impelmented by hooks and notification system)

\*DONT take this too seriously. Supporting Xcode different versions is enough now.

## XVim Application Independent Layer

This is our "virtual" Vim layer.

We implement this as if this is Vim. So the code here only have the concept of Vim.

Never include "DVTKit.h" or "NSTextView.h".

But include "XVimxxx.h"



**Evaluators** Searcher **ExCommand Options** Generate command like - Delete a line! - Jmp cursor to xxx! - Hillight here!

And tell it to layer below.

(Vim abstracted structure is here)

Vim Operation Interface

XVim Application Dependent Layer

Xcode
(Or some other applications in the future\*)

XVim Application Independent Layer

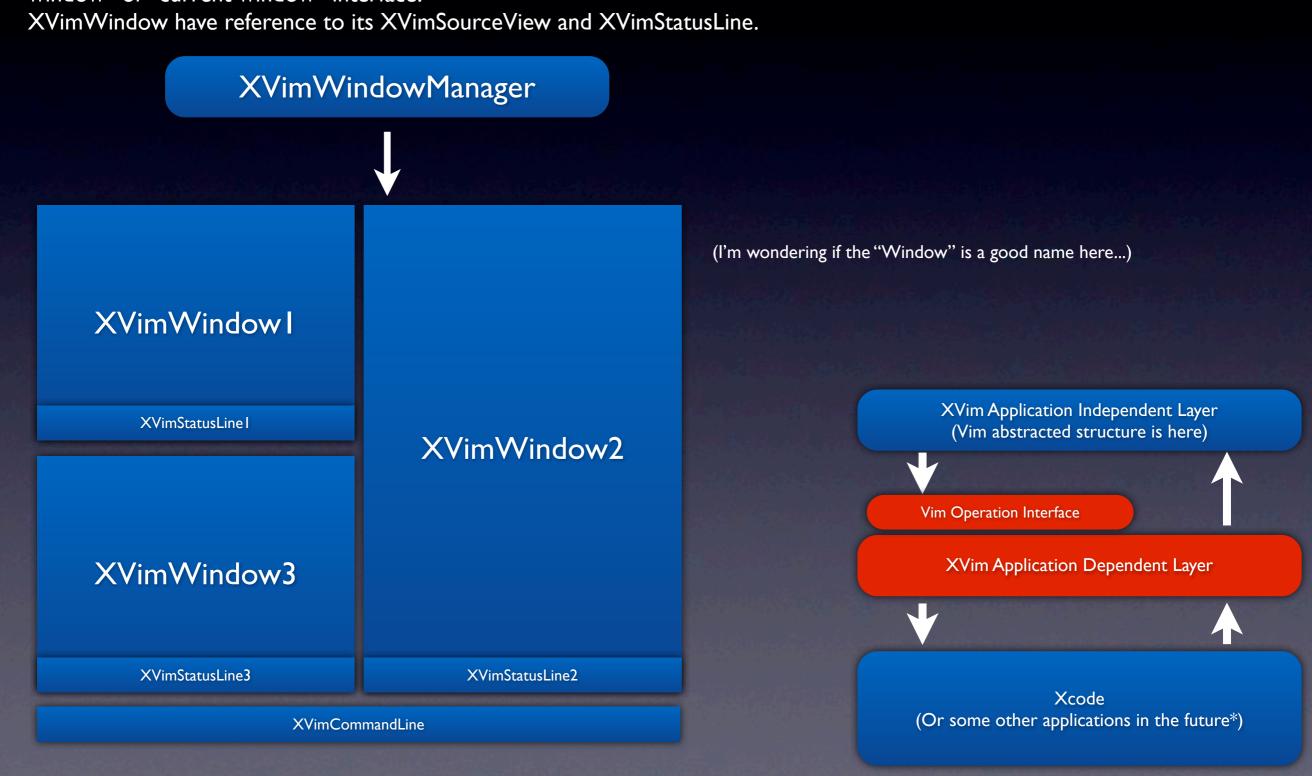
## Vim Operation Interface

This is operation interface to be sent a command from XVim Application Independent Layer.

This should be mainly XVimWindow or XVimWindowManager's interface.

The interface assumes all "virtual" Vim view structure.

XVimWindowManager manages "virtually" manages structures below. So XVimWindowManager will have "split window" or "current window" interface.



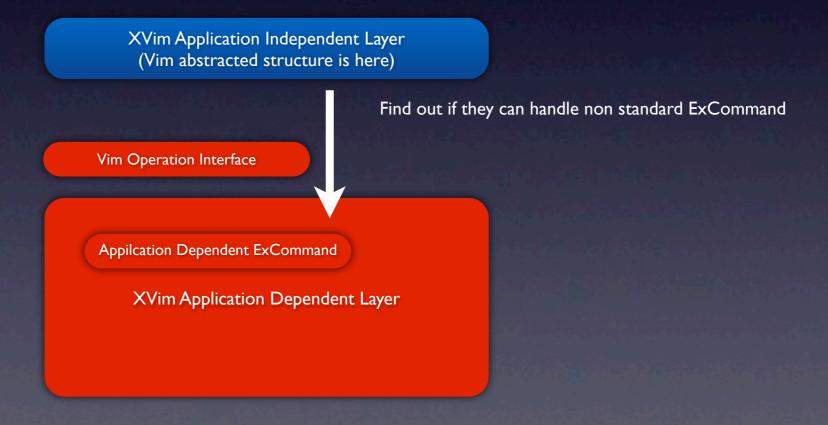
We do not have any TAB manager thing but may be introduced in the future. It will manage XVimWindowManager's collection.

Layouting windows are application dependent thing so everything about view layout is done in this layer. (Of course this layer have an interface to be called when guioptions are changed)

About Application Dependent Excommand Like ":xccmd"

This kind of command should not be included in XVim Application Independent Layer.

So XVim Operation Interface will provide the Application Dependent ExCommand executer.



## Sublayers in XVim Application Dependent Layer

Vim operations are mainly text manipulations. Delete, insert, substitute... and so on. Even these operations are implemented in XVim Application Dependent Layer most of them can be done as operation on NSTextView or NSString.

... NOT COMPLETED



