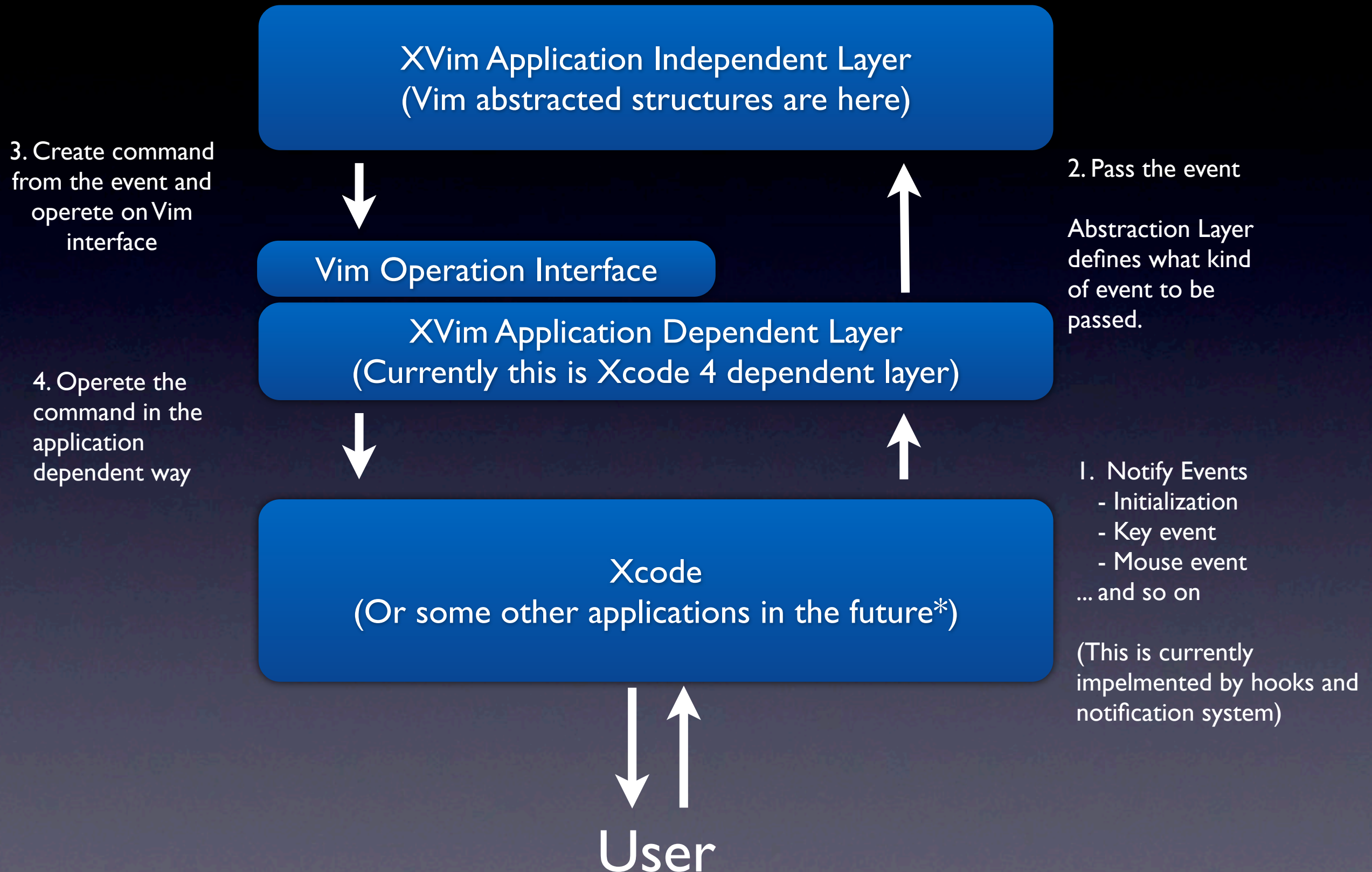


XVim Overview



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

XVim Application Independent Layer

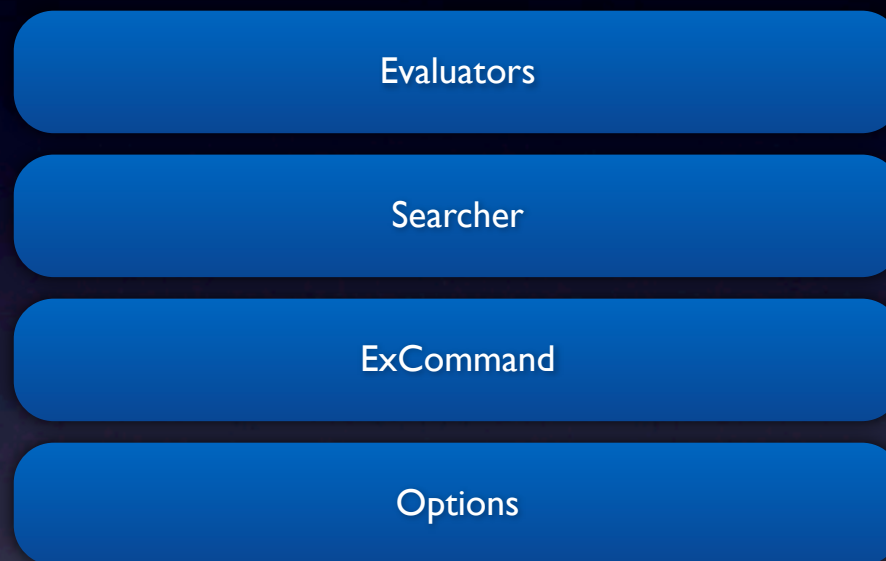
This is our “virtual” Vim layer.

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”

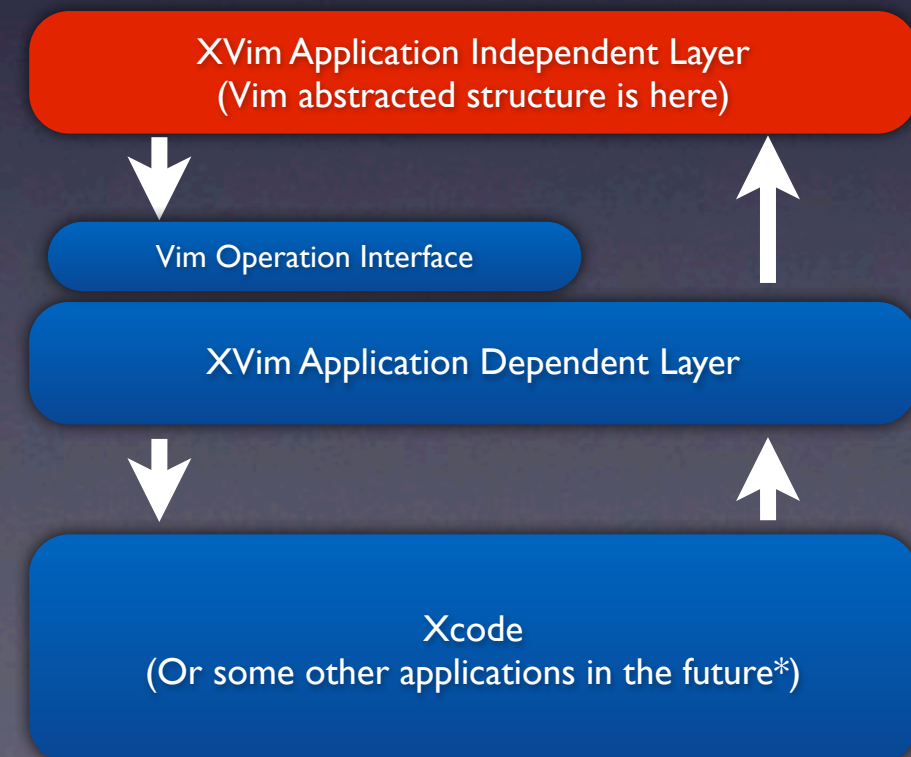
Main classes in this layer



Generate command like

- Delete a line !
- Jmp cursor to xxx!
- Hillight here !

And tell it to layer below.



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.

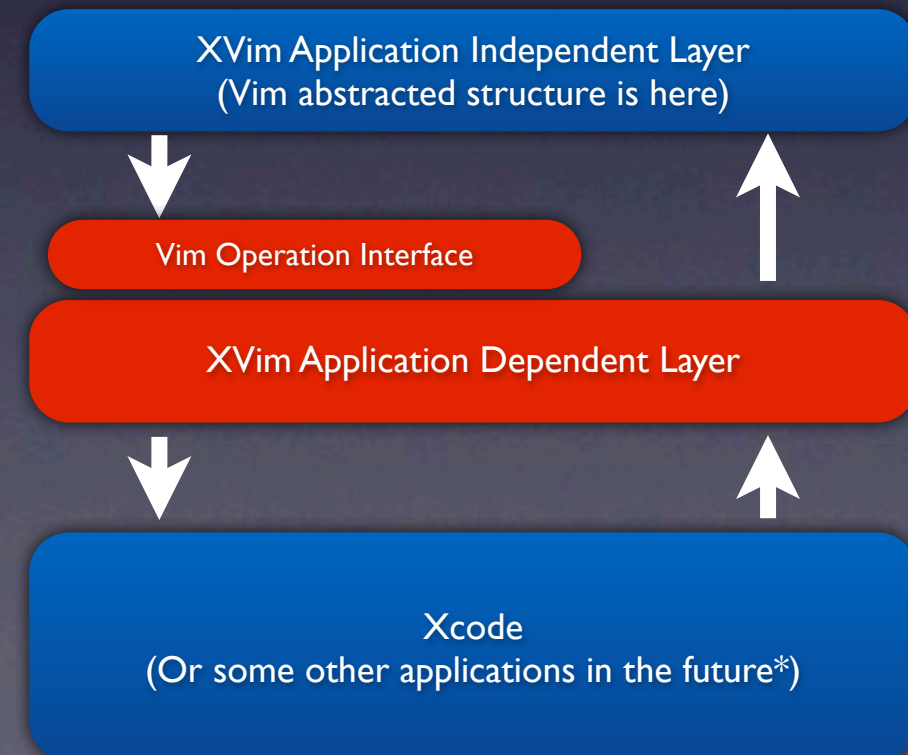
The interface assumes all "virtual" Vim view structure.

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

XVimWindow have reference to its XVimSourceView and XVimStatusLine.



(I'm wondering if the "Window" is a good name here...)

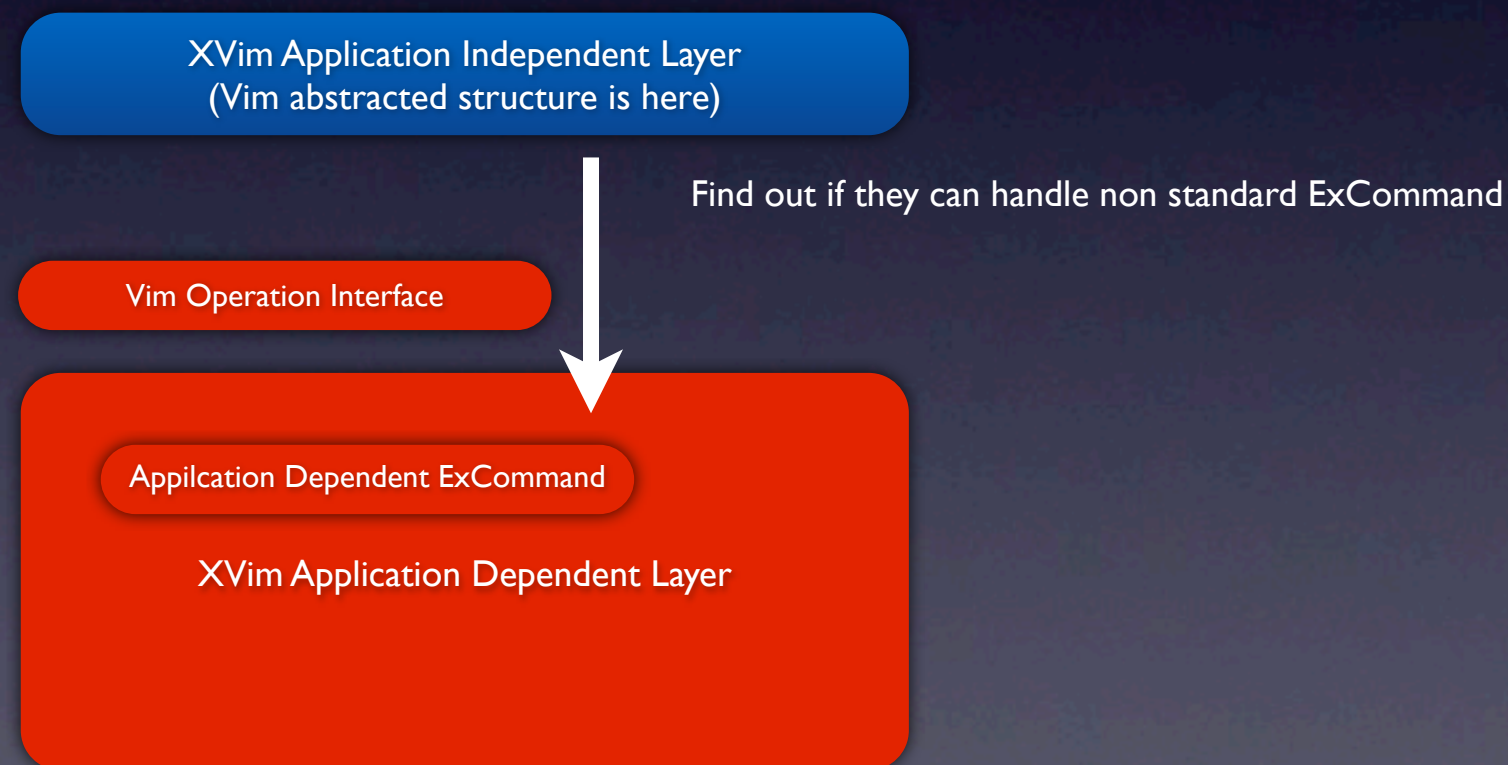


We do not have any TAB manager thing but may be introduced in the future.
It will manage XvimWindowManager's collection.

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

(I'm wondering if the "Window" is a good name here...)

