Java

1. Java SDK 6 update 32 or higher should be installed in VDI
2. Update below environment variables

JAVA\_HOME - C:\Program Files\Java\jdk1.6.0\_32

PATH - C:\Program Files\Java\jdk1.6.0\_32\bin

Eclipse setup

1. Download Eclipse Indigo (Eclipse IDE for Java EE developers) zip file and unzip

<http://www.eclipse.org/downloads/packages/release/indigo/sr2>

1. Execute eclipse.exe
2. Help > Install New Software > Work With – paste the URL

<http://m2eclipse.sonatype.org/sites/m2e/0.12.1.20110112-1712/>

1. Select Maven Integration > Next > Next > I accept.. > Finish
2. Help > Install New Software > Work With – paste the URL

<http://subclipse.tigris.org/update_1.8.x>

1. Select subclipse only > Next > I agree > finish > Restart Now (If unsigned publisher warning, click OK)
2. Open eclipse.ini and add below lines on top of the text file

-vm

C:\Program Files\Java\jdk1.6.0\_32\jre\bin\javaw.exe

1. Restart Eclipse

Subversion

1. Download and install TortoiseSVN

<http://sourceforge.net/projects/tortoisesvn/files/latest/download>

1. Right click in an empty folder > Tortoise SVN > Properties

Global ignore pattern: Append at the end : .vlt

FileVault

1. Open D:\ java\apps
2. Right Click > Tortoise SVN > Export

URL of Repository: <http://10.19.250.169/svn/repos/cba/nimbus/trunk/extra/vault-cli-2.4.26/>

Export Directory: D:\java\apps\vault-cli-2.4.26

1. D:\java\apps\vault-cli-2.4.26 should now have bin and lib folder
2. Open Environment Variables and update PATH with D:\java\apps\vault-cli-2.4.26\bin
3. To check if it is working properly, open command prompt and type vlt --help. It should show available options.

Maven repository

1. Open D:\ java\apps and put apache-maven in it.
2. D:\ java\apps\apache-maven-3.0.4 should now have bin, boot and other folder
3. Open Environment Variables and update PATH with D:\java\apps\apache-maven-3.0.4\bin and also set M2\_HOME with D:\java\apps\apache-maven-3.0.4
4. To check if it is working properly, open command prompt and type mvn --version. It should show maven version.
5. Open Eclipse
6. Windows > Preferences > Maven > User settings – Click on browse and put D:\java\apps\apache-maven-3.0.4\conf\settings.xml and click ok

CQ Jars

1. Open D:\java\apps\cq\_jars
2. Right Click > Tortoise SVN > Export

URL of Repository: <http://10.19.250.169/svn/repos/cba/nimbus/trunk/extra/cq_jars/>

Export Directory: D:\java\apps\cq\_jars

1. D:\java\apps\cq\_jars should now have jar files

Get latest code from subversion repository

1. In D:\ create a folder, java
2. Inside java, create folder cq5\_project
3. Inside D:\java\cq5\_project, Right Click > SVN Checkout

URL: <svn url>

Checkout directory: D:\java\cq5\_project

CRXDE

1. Download CRXDE editor from <https://www.day.com/day/en/products/crx/download/downloadcrxde.html>

Project setup in Eclipse

1. Open eclipse
2. File > Import > Maven > Existing Maven Projects > Next

Root Directory – Browse - D:\java\cq5\_project

1. Finish
2. Right click on repo > Build Path > Configure build path > Libraries
3. Select – Add external jar – browse to D:\cq\_jars – Ctrl + A (select all) –open – ok

Project setup in CRXDE

1. Open CRXDE
2. Enter <http://localhost:4502> and your credentials
3. The editor will show libs, etc and apps folder

Initial setup

1. Start a new instance of CQ 5 in port 4502
2. Get the cq5.1 update and mobile-emulator package from svn

In an empty folder,

Right Click > Tortoise SVN > Export

URL of Repository: <http://10.19.250.169/svn/repos/cba/nimbus/trunk/extra/cq_update_package/>

1. Now import both cq-update-pkg-5.5-1-load5.zip and apple-device-emulators.zip in your local instance using package manager

Initial syncing of code from svn to local instance

1. Go to D:\java\cq5\_project\repo in command prompt. And run vltclean.bat. This will compile the java code and will make bundle
2. Now go to D:\java\cq5\_project\repo in command prompt and execute vltCheck-Out.bat