Source Control Proven Practises

Branching

Plan your branching

Your branching will only ever be as good as your planning. Spend the time to plan.

Own the merge

Perfect auto merging branches is a fairy tale. When you merge, take responsibility and own the merge.

Look after branches

Branches are like small birds. Each branch takes time & money, so be prepared to look after it.

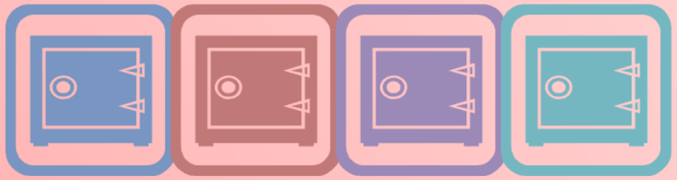
Repository

Don't break the build

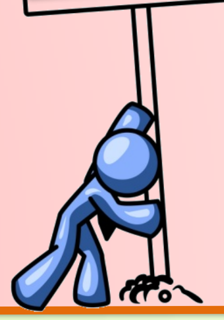
The code in the trunk should always build – it's the heart beat of your project. If you break the build you should sit in the corner.

Separate your repo

Don't have a single rep for all your work. If it's not directly needed, put it in a separate repository.



Use labels



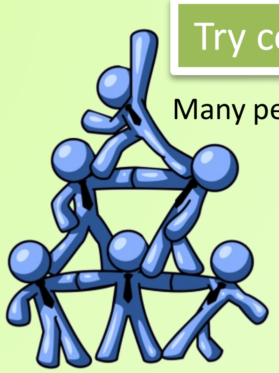
Use labels to identify key points in your project.

Don't forget to shelve

If you need to check in code & which is not ready to be built: then shelve the changes!

Technology

Try concurrent access



Many people are scared of concurrent access and shouldn't be. Try it out – you can get a great productivity boost.

Don't be afraid of branching concepts

Branching & merging is less complex and gives you more than the alternatives. Rather be scared of the alternatives.

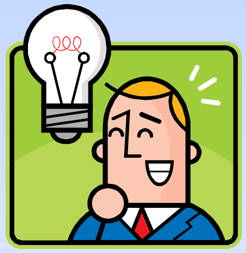
Auto merge is for check out only

NEVER auto merge in check-in, it's just a way to break everything.

Management

Useful & meaningful check in messages

Check in messages give you and everyone else a way to find out what changed. Without them expect to read lots of other peoples code.



Don't use the audit trail for blame

The audit trail is never for assigning blame. If you do, you'll guarantee that adoption of your source control will fail. If you want adoption use it for highlighting great work.

You

Keep up to date

The longer you are out of date with the repo, the bigger the pain to catch up.

Be light and quick with checkouts

Checkout the minimum and for the shortest time. It keeps the environment clean, agile and healthy.

Don't check in unneeded binaries

Source control is for source code and reference binaries which are needed to compile your source code. Keep the compiled version of your code on your machine.

Working folders should be disposable

The health check for your setup is that the easier it is to delete your working folder, the healthier you are.

Use undo/revert sparingly

It's the quickest way to lose code.

BARONE, BUDGE & DOMINICK
Business • Technology • Solutions

Source Control is a easy to get right when you follow these proven practises.

Source Control