Transactional editing of project/solution

Sometimes during the progress of development you have an idea that you should try, so you must do a lot of modifications on the files of the project/solution.

Naturally a vast majority of these rounds is useful and it may persist, but there are times when this modification process is an impasse. If you have made only a few modifications you can rollback by a series of „undo" commands. But if these modifications contain inserted or deleted files or the amount of modifications is too much the "undo" isn't a good solution.

You can use a version control system for tracking changes (for example GIT) but this process is too complicated for several situations (typically when the number of changes is not too high or only one developer is involved).

Here is a plan for a new operation of editor or IDE:

- Open a projekt/solution
- Switch on „transactions of project files" flag
- Auto start a transaction
- modify files of the project and/or insert or delete files
- compile and run, debug these modifications
- when finishing the modification process you can commit modifications or rollback them
- a new transaction will start again in case you do not switch off „transactions of project files"

Others:

The simple way for the editor/IDE to work is when a transaction lives only during the actual run of the program. An uncommited transaction asks if you would like to commit/rollback, like unsaved files.

A more complicated way for the editor/IDE to work is when there is a third option for a not closed transaction. This option can save the transaction and at the next start of the editor it can continue the last transaction.

A realization:

When a trancaction is switched on, the modification of the file checks the affected file. If it is not saved yet the editor/IDE saves it. When the transaction is commited all saves are droped out. When rollback happens these saves will all be copied back.