

Versioning and authenticity

source UK Data Archive¹

The strategy you adopt in order to keep track of the versions you keep of your document depends on the number of users, the locations where the data is stored and the need for synchronisation.

The version of a file can be identified via:

- ✓ date in file (name)
- ✓ version numbering in file name (v1, v2, v3 or 00.01, 01.00)
- ✓ file history / version control table / notes in a file
- ✓ recording versions, dates, authors and details of changes to the file

Version control can also be maintained through:

- ✓ version control facilities within software (Ms Word)
- ✓ versioning software, e.g. Subversion (SVN), GIT
- ✓ controlling rights to file-editing
- ✓ manual merging of entries or edits by multiple users
- ✓ file sharing services with integrated version control (f.i. Sharepoint)

Best practice is to:

- ✓ Decide on versions to keep, for how long and how to organise versions (milestone : major instead of minor, e.g 02-00 but not 02-01)
- ✓ uniquely identify files using a systematic naming convention
- ✓ record changes made to a file when a new version is created
- ✓ record relationships between items where needed, e.g. relationship between code and the data file it is run against; between data file and related documentation or metadata; or between multiple files
- ✓ track the location of files if they are stored in a variety of locations
- ✓ regularly synchronise files in different locations (e.g. using special software like MS SyncToy software)
- ✓ identify a single location for the storage of milestone and master versions

¹ Based on : *Managing and sharing research data : a guide to good practice* / Louise Corti, Veerle Van den Eynden, et al., Los Angeles : Sage, 2014

Authenticity

It is important to be able to demonstrate the **authenticity** of data and to be able to prevent **unauthorised access** to data that may potentially lead to unauthorised changes.

Best practice to ensure authenticity is to:

- ✓ save files with a new version number *before* editing
- ✓ keep a single **master file** of data
 - ✓ assign responsibility for master files to a single project team member
 - ✓ regulate write access to master versions of data files
 - ✓ record all changes to master files
 - ✓ maintain old master files in case later ones contain errors
 - ✓ archive copies of master files at regular intervals
 - ✓ develop a formal procedure for the destruction of master files