Data Handling for Researchers

Christian Jacobs, Matthew Piggott, Gerard Gorman, David Ham

4 March 2014

Course Outcomes

- Understand what data is and why it is important.
- Understand the need to backup, compress, and encrypt data.
- Be aware of best practices.
- Be aware of the tools available for analysis and testing.
- Know the basics of version-controlled file repositories.

What is data?

Definition

- Data is a set of values corresponding to one or more quantitative or qualitative variables.

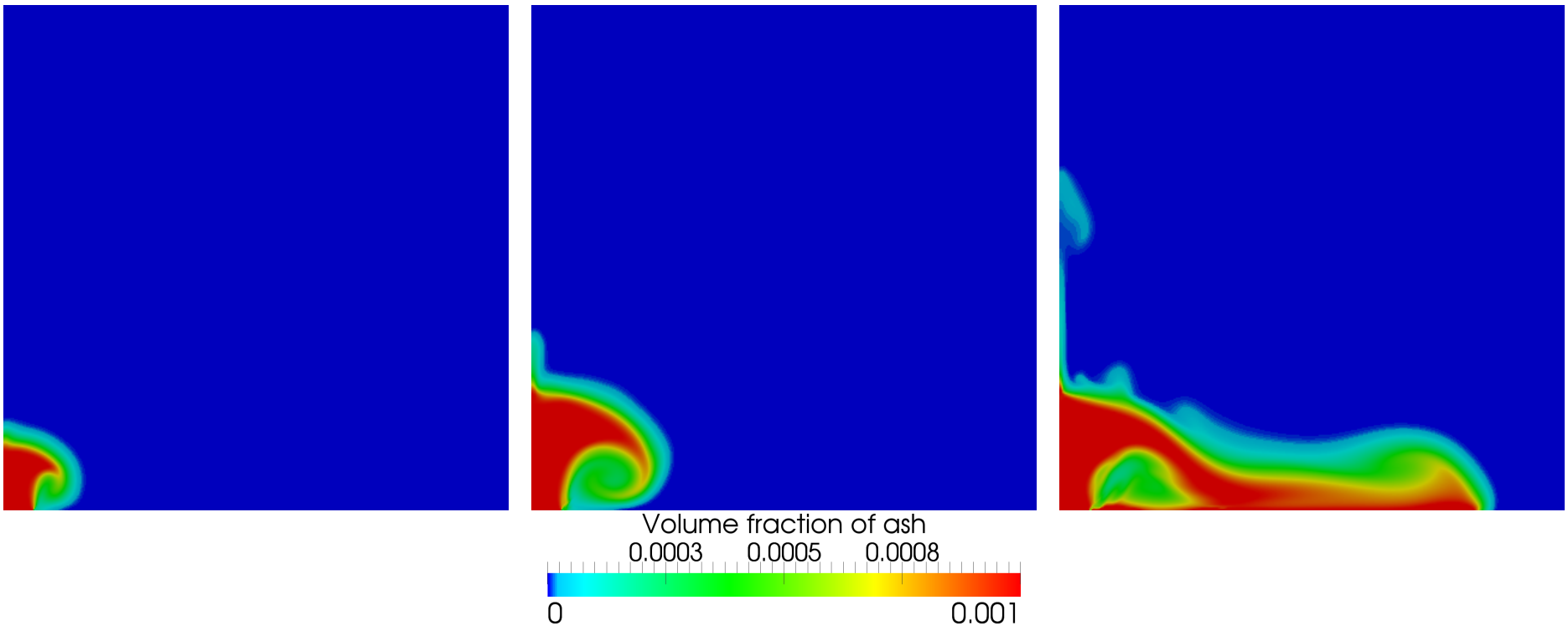
- Examples:
  - Sea levels measured every hour at a fixed location
  - Speed of a car throughout time
  - Metadata (= data that describes other data) for webpages
  - Wind velocity at different locations in the UK
  - Depth of a particle settling in a water tank, measured at various times.

- Data can come from existing sources, may be derived from several data sets, or a new independent data set can be generated.

What is data?

More examples

Values of particle concentration in space following a volcanic eruption:



What is data?

More examples

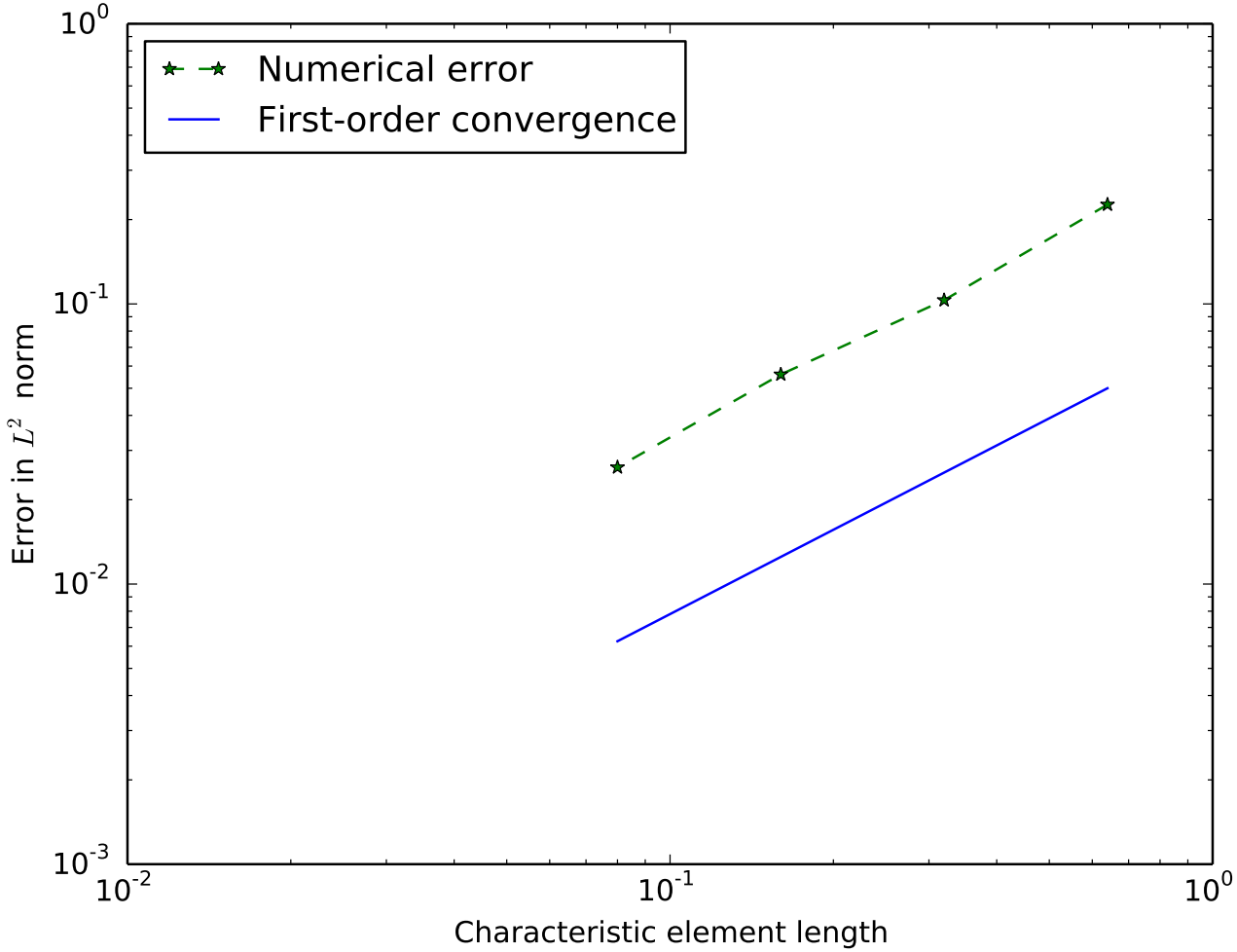
Air density at various temperatures. Data from the Density page on Wikipedia.

#DEG_C	DENSITY
-10	1.341
-5	1.316
0	1.293
5	1.269
10	1.247
15	1.225
20	1.204
25	1.184
30	1.164

What is data?

More examples

Numerical solution error against grid spacing:



Why is data important?

- Allows new scientific discoveries to be made.
- Journals and research councils are encouraging the sharing of data to:
  - promote research output,
  - minimise the duplication of data,
  - increase transparency and accountability,
  - allow fellow researchers to scrutinise and evaluate the data.
- The effective handling and management of all research data plays an important role in each of these processes.

Why is data important?

Extracts from the "Notice of Retraction" for Bertoia et al. (2011) in *Hypertension*:

"For the article by Bertoia et al, "Implications of New Hypertension Guidelines in the United States," [...] the authors discovered an error in the code for analyzing the data.".

"Consequently, the sample size was twice as large as it should have been (24989 instead of 10198)."

"For these reasons, *Hypertension* requested that the authors resubmit a corrected version of this manuscript."

doi: 10.1161/HYP.0b013e318269bc7a

Why is data important?

"By reanalysing inaccurately presented data of Kerr et al. (2006), we refute their claims that area-corrected species richness of endemic Madagascan birds and mammals increases toward the Equator and is best explained by environmental factors, and that the rainforest mid-domain effect (MDE) Lees et al. (1999) demonstrated is artefactual."

Lees and Colwell (2007). doi: 10.1111/j.1461-0248.2007.01040.x

Issues to consider

Data provenance

- Where did the data originally come from? Is there a chain that can be traced back to the origin of the data?
- Can it be trusted? (Is the author list available? Reputable journal?)
- Is it reproducible?
- Has the data already been used successfully? Any reported issues?

Issues to consider

Licensing

- Who can use the data, and how?
- Who owns the copyright? Are you allowed to publish it or use it in your thesis?
- Any user licence? Creative Commons licences are becoming more popular and offer more freedom.
- Data produced when employed by a Government agency may be under Crown Copyright. The copyright does not belong to an individual, and is instead under the control of Her Majesty's Stationery Office (HMSO) - see `www.nationalarchives.gov.uk/information-management`.

- **You need to know the answer to these points before you base any of your work on this data.**

Issues to consider

File formats

- Using standardised, open-source file formats makes your data portable (between computers and operating systems) and facilitates sharing of data by other researchers.
- Comma-Separated Value (CSV): a commonly-used format for simple data sets. Values in a single row are separated by commas. CSV files can contain multiple rows, thereby forming a table.
- eXtensible Markup Language (XML): each piece of data is encapsulated in a tag which annotates/describes it.
- Network Common Data Form (NetCDF): commonly used in numerical climate and ocean models.

Issues to consider

File formats – CSV

```
# First column: time
# Next 3 columns: FreeSurfacePerturbation, Velocity x-component, Velocity y-component from the first detector
# Next 3 columns, " " " from the second detector.

0.0, 0.0, 1e-16, 1e-16, 0.0, 1e-16, 1e-16
100.0, 8.99503226119e-10, 5.02086158045e-10, 3.69529691258e-14, 8.00940144499e-10, 4.55592822082e-10, 3.31913461237e-15
200.0, 1.13283393532e-08, 4.53350070317e-09, 8.50895803716e-14, 1.01512878681e-08, 4.11263071875e-09, -2.66498860334e-15
```

Issues to consider

File formats – XML

```
<timestepping>
  <current_time>
    <real_value rank="0">0</real_value>
  </current_time>
  <timestep>
    <real_value rank="0">0.0001</real_value>
  </timestep>
  <finish_time>
    <real_value rank="0">600.0</real_value>
  </finish_time>
  <nonlinear_iterations>
    <integer_value rank="0">2</integer_value>
    <tolerance>
      <real_value rank="0">1.0e-12</real_value>
      <infinity_norm/>
    </tolerance>
  </nonlinear_iterations>
</timestepping>
```

Issues to consider

File formats – NetCDF

- A binary file format commonly used in numerical climate and ocean models.
- It is "self-describing": header information and metadata are automatically included.
- Several NetCDF file readers are readily available.
- `www.unidata.ucar.edu/software/netcdf`

Issues to consider

Storage options

- Optical media (CDs 700 MB, DVDs 4.7 GB, Blu-ray 25+ GB) and flash drives - for small files e.g. presentations, theses, papers.
- Magnetic media (hard drives) - for larger files (e.g. simulation output).
- Cloud services (e.g. Dropbox, Google Drive).
- Always maintain a good file hierarchy and naming convention when storing data files.

Issues to consider

Backing up

- **The importance of regularly backing up data cannot be stressed enough!**
- What if your hard drive failed right now? What if your computer (and any connected backup device) was stolen?
- Storage space is reasonably cheap.
- Always keep several regular backups, far apart from each other (not in the same building).
- Know the Imperial College data backup policy.
- imperial.ac.uk/ict/services/computerroom/file_and_backup_services

Issues to consider

Encryption

- Be aware of responsibilities to encrypt sensitive information.
- Encrypting emails: Pretty Good Privacy (PGP) keys. `www.pgp.com`
- Encrypting hard drives: TrueCrypt (Windows, Linux, Mac OS), eCryptfs (Linux).

- The 'Climatic Research Unit email controversy': "The Climatic Research Unit email controversy (also known as "Climategate")[2][3] began in November 2009 with the hacking of a server at the Climatic Research Unit (CRU) at the University of East Anglia (UEA) by an external attacker. [...] Climate change critics and others denying the significance of human caused climate change argued that the emails showed that global warming was a scientific conspiracy, in which they alleged that scientists manipulated climate data and attempted to suppress critics." http://en.wikipedia.org/wiki/Climatic_Research_Unit_email_controversy

Issues to consider

Big Data

- Big Data is one of the key challenges in data science.
- Involves data sets that are extremely large, thereby creating additional difficulties when analysing them.
- Need novel and efficient tools to help tackle this issue.
- Data Science Institute at Imperial.

Creating and manipulating data

Tools

- Data sets are often merged, manipulated, and generated using computer programs or scripts.
- Often written in MATLAB or Python.

Creating and manipulating data

Source code examples

Example from `www.programming4scientists.com`:

```
FUNCTION comppoly(x)
float y1, y2
float  a1=0.1,  b1=0.3,  a2=2.1,
b2=5.3, c=0.22
y1 = a1*x + b1
y2 = a1*x^2 + b2*x + c
return(y2>y1)
END FUNCTION
```

```
FUNCTION ComparePolynomials(x)
//DECLARE VARIABLES, PARAMETERS
float y_line, y_quadratic

float lineParam = [0.1, 0.3]
float quadParam = [2.1, 5.3, 0.22]

//CALCULATE THE LINE AND QUADRATIC
VALUES AT X
y_line = lineParam[0]*x + lineParam[1]
y_quadratic = quadParam[0]*x^2 + quadParam[1]*x
+ quadParam[2]

//COMPARE THE FUNCTIONS, RETURNING
A LOGICAL
return(y_line > y_quadratic)
END FUNCTION
```

- Always comment and document your code to help yourself and others to understand how to use it.
- Use sensible variable names.
- Use metadata to document the name of the author, the date the data was created, any terms of use, etc.

Creating and manipulating data

Quality assurance

- Test programs for correctness in order to have confidence in the results.
- Regression testing - identifies new faults that have been introduced from changes to the code.
- Programs can break even without changes to the source code (e.g. compiler faults).
- The data itself should also be tested using verification and validation techniques.

Creating and manipulating data

Quality assurance – Buildbot

Buildbot (`buildbot.net`): An automated continuous testing framework. The code is tested after any change is made.

Creating and manipulating data

Version control systems

- Used to keep track of changes to data, and the programs used to generate data.
- Facilitates team development.

Using 'u' rather than 'u_old' in the drag term. Browse code

Also enabled the detectors, which might cause things to fail on buildbot since vtktools.py is not currently available in Firedrake.

master

ctjacobs authored 12 days ago 1 parent cb5b7a2 commit 04d90aa9f63e631a9bf8b65428b0843f90a52d31

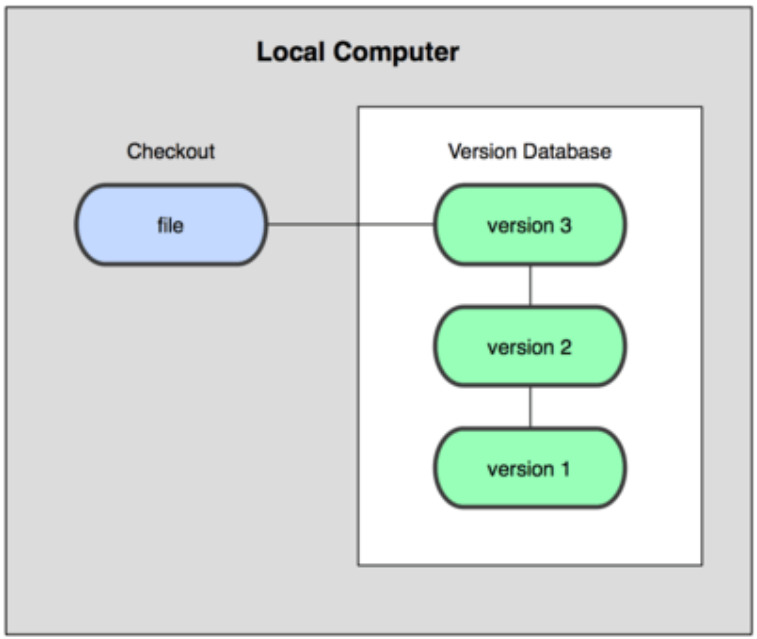
Showing **1 changed file** with **20 additions** and **12 deletions**. Show Diff Stats

```
@@ -289,7 +297,7 @@ def run(self):
289 297             D_momentum = 0
290 298             magnitude = 0
291 299             for dim in range(dimension):
292     -               magnitude += dot(self.u_old[dim], self.u_old[dim])
    300 +               magnitude += dot(self.u[dim], self.u[dim])
293 301             magnitude = sqrt(magnitude)
294 302             for dim in range(dimension):
295 303                 D_momentum += -inner(self.w[dim], (C_D*magnitude/(self.h_mean + self.h))*self.u[dim])*dx
```

Creating and manipulating data

Version control systems — Motivation

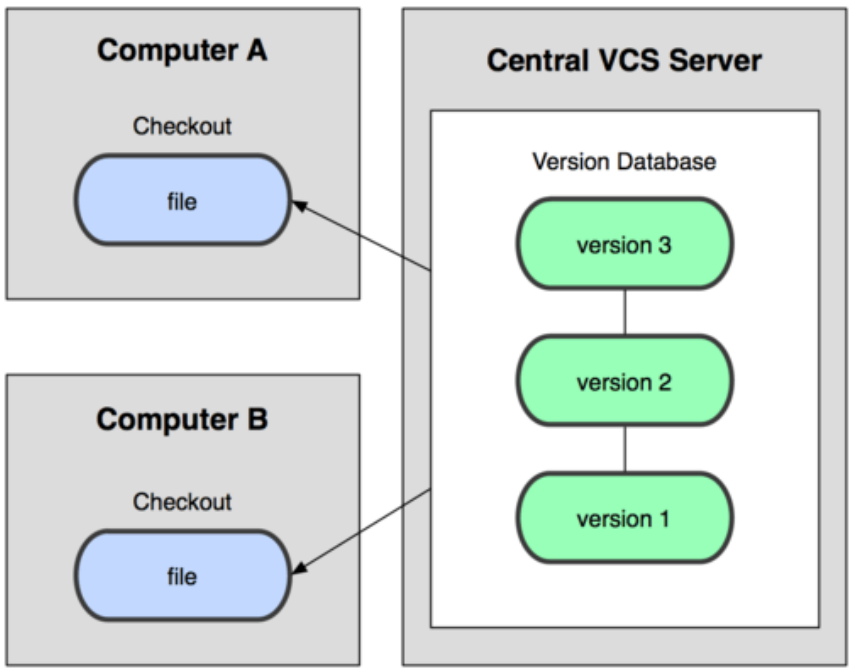
- Many people choose to keep track of the different versions of a file by simply copying the current state of the file into a separate folder each time they want to record that change.
- This is very error prone. Easy to overwrite files accidentally or forgetting to make a note of what exactly was changed. Extra work to merge different changes together.



Creating and manipulating data

Version control systems — Centralised

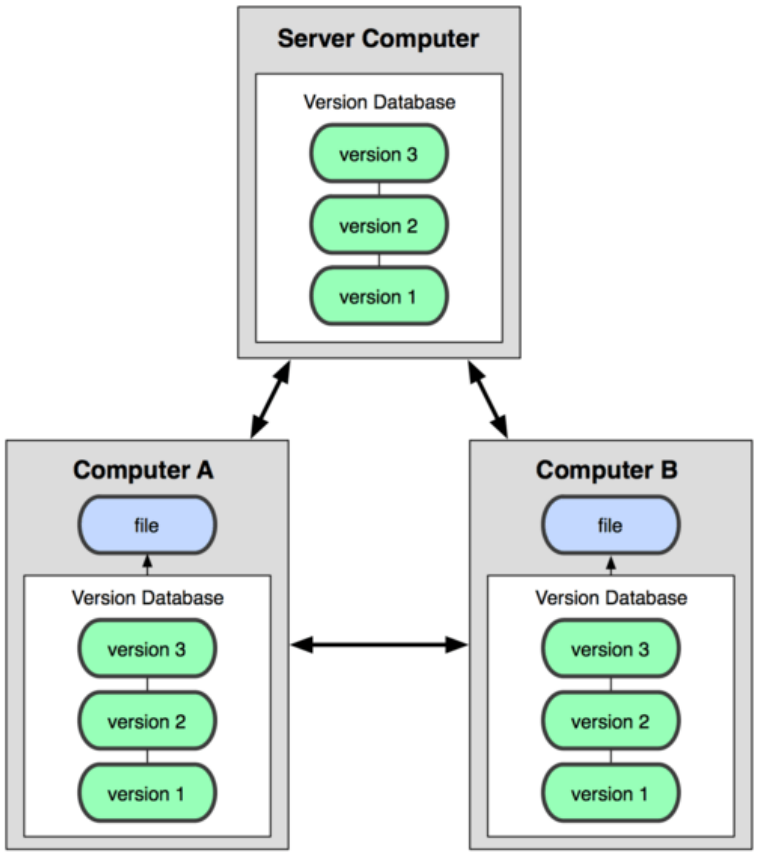
- A server stores all the files and the version history.
- Each time the file's state is saved ('committed'), a log message must be written. This also gets stored on the server.
- Useful when multiple people are working on the same file.
- Requires a connection to the server to commit changes.



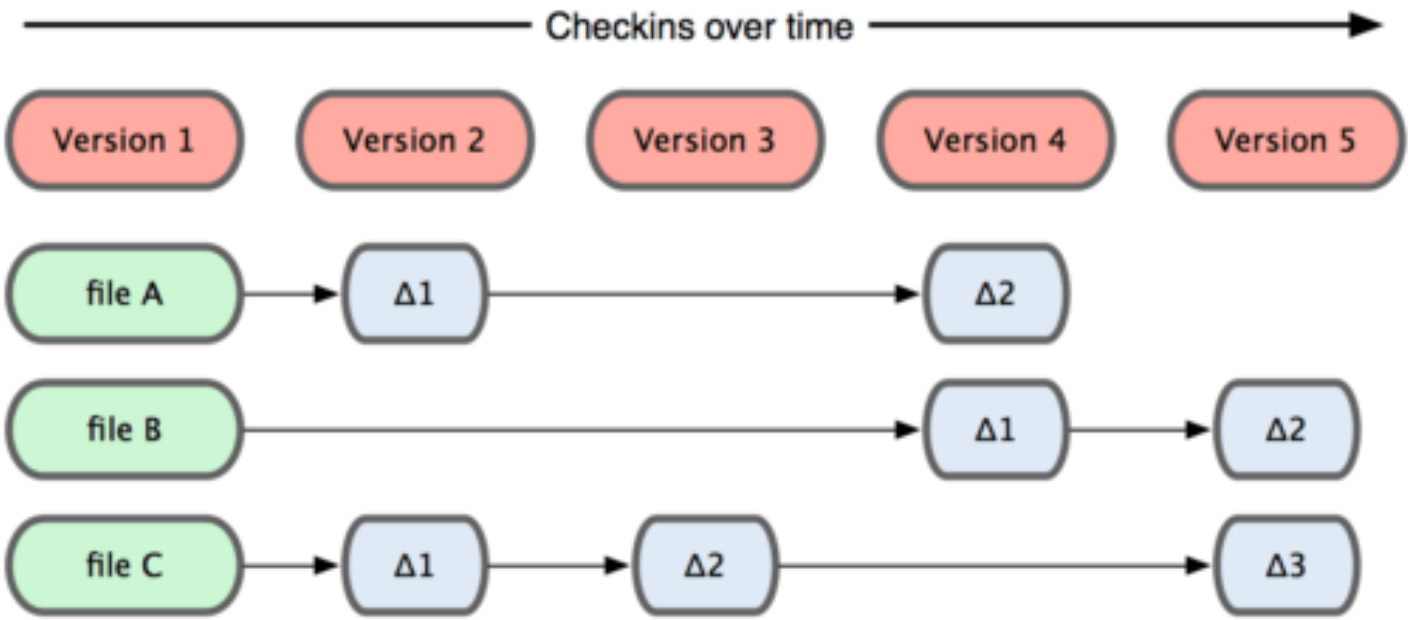
Creating and manipulating data

Version control systems — Distributed

- The checkout is effectively a full backup of the data.
- Does not require a connection to the server to commit changes.



- When committing or 'checking-in' changes, the differences between the files you are committing and the files from the previous version are recorded in the history log.



- Examples: Subversion (`subversion.apache.org`), Bazaar (`bazaar.canonical.com`), Git (`git-scm.com`)
- Some services such as GitHub (`www.github.com`) and Bitbucket (`www.bitbucket.org`) offer free Git-based repositories.

Creating and manipulating data

Exercise 1

- Let's work our way through a set of Git exercises created by GitHub: `http://try.github.io`

- Initialise a verson-controlled repository using `git init`
- Add files to the repository using `git add <file_name_here>`
- Remove files from the repository using `git rm <file_name_here>`
- Commit any changes using `git commit -a` (the `-a` adds the modified files to the staging area, and then commits the changes - see the documentation for more information.)

Creating and manipulating data

Exercise 2

- Set up an account at `www.github.com`.
- Download the Git for Windows tool here: `windows.github.com`
- Set up a new repository called `data-handling-course`.
- Download the files from the following web address to your repository's folder (Desktop/GitHub/data-handling-course): amcg.ese.ic.ac.uk/∼ctj10/data-handling-course
- Run the program `plot_rainfall.m`. A plot of the mean rainfall will be saved as `rainfall_plot.png`. Add and commit this file to your data-handling-course repository.
- Run the regression test program `test_rainfall.m`.

- Note 1: Any work that you store in a GitHub repository is made public (unless you pay for a private repository).
- Note 2: As an alternative, Bitbucket offers unlimited free private repositories.
- Note 3: You can delete the GitHub account at any time under the account settings page.

Creating and manipulating data

Exercise 2 - Continued

- If you need additional functionality: Git for Windows (`msysgit.github.io`), or gitk for Linux (`git-scm.com/docs/gitk`).
- These aren't tied to just GitHub or Bitbucket.

Additional resources

- The UK Data Archive: `www.data-archive.ac.uk`
- The Software Sustainability Institute: `www.software.ac.uk`
- Software Carpentry: `software-carpentry.org`
- Digital Curation Centre: `www.dcc.ac.uk`
- Information Commissioner's Office: `ico.org.uk`
- IASSIST: `www.iassistdata.org`

- G. Wilson et al. (2014). Best Practices for Scientific Computing, PLoS Biol 12(1). doi: 10.1371/journal.pbio.1001745
- L. Hatton, A. Roberts (1994). How accurate is scientific software?, Software Engineering, IEEE Transactions, 20(10):785–797. doi: 10.1109/32.328993