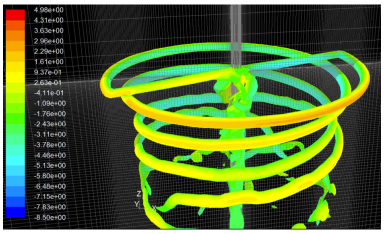
UNIVERSITY OF Southampton

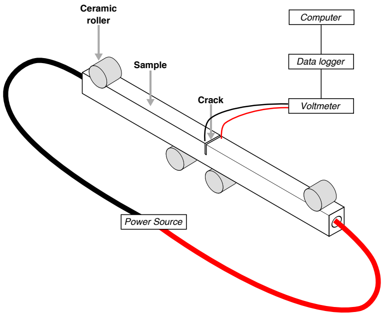
Introducing Research Data

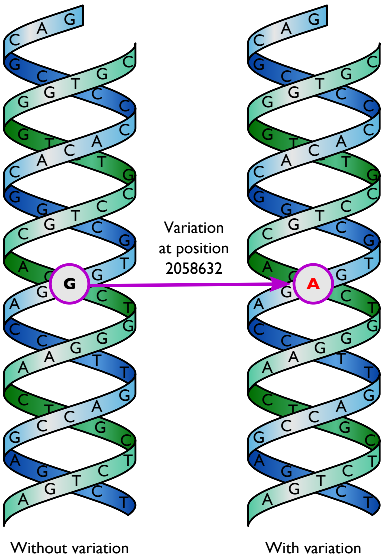
Edited by Mark Scott and Simon Cox

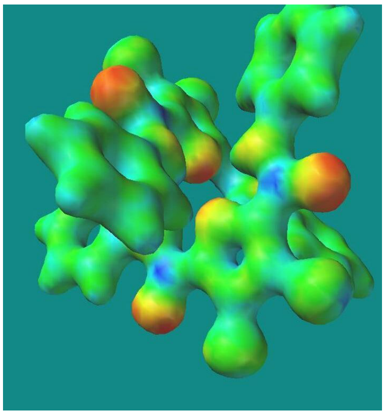
University of Southampton

United Kingdom









Introducing Research Data

Overall content by Mark Scott, Richard Boardman, Philippa Reed and Simon Cox in the Faculty of Engineering and the Environment.

Second and third editions revised by Mark Scott, Dorothy Byatt and Simon Cox.

Fourth edition revised by Mark Scott, Isobel Stark, Dorothy Byatt and Simon Cox.

Case studies produced with help from Andy Collins, Thomas Mbuya, Kath Soady, Gregory Jasion, Simon Coles and Graeme Earl.

This is revision 160:da6022521420 of this document, created 2016-10-31 .

Introducing Research Data

Every discipline, from the arts and humanities to physics, is increasingly using data to drive forward its goals. Medicine might use it for recording the statistics of a particular drugs trial; physicists have complex experiments – such as the the Large Hadron Collider at CERN – producing massive quantities of data on an hourly basis; and archaeologists meticulously preserve digital records of excavation sites and artefacts.

Data comes in many forms: small, large, simple, complex and colourful. This guide first introduces the forms data can take by showing five ways of looking at data, then presents some case studies of data usage in several disciplines in an attempt to illustrate the types of data you might encounter in your research and give you some tips or tricks that will help you in your own discipline.

The final part of the guide gives some general advice on managing and understanding data.

Contents

Part I
Five Ways To Think About Research Data

Science has progressed by 'standing on the shoulders of giants' and for centuries research and knowledge has been shared through the publication and disemmination of books, papers and scholarly communications. Moving forward much of our understanding builds on (large scale) data sets which have been collected or generated as part of this scientific process of discovery. How will this be made available for future generations? How will we ensure that, once collected or generated, others can stand on the shoulders of the data we produce?

Deciding on how to look after data depends on what your data looks like and what needs to be done with it. You should find out if your discipline already has standard practices and use them. We hope that this brief introduction will give some templates of what is already being done in a few disciplines and enable you to start thinking about what you might do with your research data to make it accessible to others.

Further University of Southampton guidance can be found on the library's web site `http://library.soton.ac.uk/researchdata`. Any research data management questions can be emailed to `researchdata@soton.ac.uk`.

This part of the guide introduces five ways of looking at research data.

1 Research data collection

The first way of thinking about research data is where it comes from (Research Information Network, 2008). Each of the case studies in Part II illustrates one of these categories.

Reference data: *Example: the reference human genome sequence in Case Study 1*
A data set that can be used for validation, comparison or information lookup.

Scientific experiments: *Example: materials engineering fatigue test in Case Study 2*
Data generated by, e.g. instruments during a scientific experiment.

Models or simulations: *Example: CFD helicopter rotor wake simulation in Case Study 3*
Data generated on computer by an algorithm, mathematical model, or the simulation of an experiment. A computer simulation can help when experiments are too expensive, time consuming, dangerous or even impossible to perform.

Derived data: *Example: chemical structures in chemistry in Case Study 4*
A data set created by taking existing data and performing some manipulation to it. Each data set requires careful curation because the original data may be needed to understand the new data.

Observations: *Example: archaeological dig in Case Study 5*
Data generated by recording observations of a specific, possibly unrepeatable, event at a specific time or location.

2 Types of research data

Research can come in many different forms, some electronic and some physical. Here are some examples:

- Electronic text documents, e.g. text, PDF, Microsoft Word files
- Spreadsheets
- Laboratory notebooks, field notebooks and diaries
- Questionnaires, transcripts and codebooks
- Audiotapes and videotapes
- Photographs and films
- Examination results
- Specimens, samples, artefacts and slides
- Digital objects, e.g. figures, videos
- Database schemas
- Database contents
- Models, algorithms and scripts
- Software configuration, e.g. case files
- Software pre-process files, e.g. geometry, mesh
- Software post-process files, e.g. plots, comma-separated value data (CSV)
- Methodologies, workflows, standard operating procedures and protocols
- Experimental results
- Metadata (data describing data), e.g. environmental conditions during experiment
- Other data files, e.g. literature review records, email archives

3 Electronic storage

The third way to think about research data is how it is stored on a computer. Here are some of the categories of electronic data:

Textual, e.g.:

- Flat text files
- Microsoft Word
- PDF
- RTF

Numerical, e.g.:

- Excel
- CSV

Multimedia, e.g.:

- Image (JPEG, TIFF, DICOM)
- Movie (MPEG, AVI)
- Audio (MP3, WAV, OGG)

Structured, e.g.:

- Multi-purpose (XML)
- Relational (MySQL database)

Software code, e.g.:

- Java
- C

Software specific, e.g.:

- Mesh
- Geometry
- 3D CAD
- Statistical model

Discipline specific, e.g.:

- Flexible Image Transport System (FITS) in astronomy
- Crystallographic Information File (CIF) in chemistry

Instrument specific, e.g.:

- Olympus Confocal Microscope Data Format
- Carl Zeiss Digital Microscopic Image Format (ZVI)

Data can be born digitally, such as a simulation, or ingested into a computer, such as scanning a photograph. Some data can remain in a non-digital format.

4 Size and complexity of data sets

Another consideration when evaluating research data is the size of the files. These are subjective, e.g. a set of photographs may be considered to be large to that researcher, but another researcher may work with three dimensional X-ray data which can be many times larger.

- Individual large file, e.g. database; virtual machine's hard disk; raw CT data; movie
- Set of small files, collectively large, e.g. time steps in CFD simulation (collectively representing the full simulation, but individually a subset of time which can be processed separately); individual frames of movie
- Set of small files, collectively small, e.g. source code where the entire set of files are needed to compile
- Individual small file, e.g. CSV files produced by a numerical solver; photograph
- Combinations of the above, e.g. a large file accompanied by a small text file describing its contents.

Figure 1: Examples of data sets, and their sizes and complexities

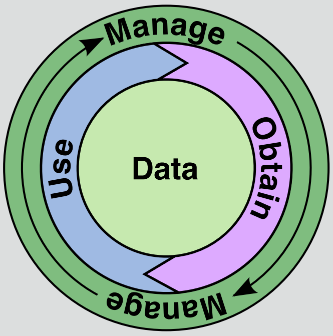
Complexity	Type of data	Size
Individual file	Raw CT data	10–100s of gigabytes
	Video	Gigabytes
	Photograph	Megabytes
Set of files	Individual frames of a movie	Gigabytes
	Source code files	Kilobytes/megabytes

5 Data life cycle

During its lifetime, data goes through a number of phases. Different disciplines have different ways of thinking about this life cycle as can be seen by the figures in this section.

The first two figures show a high-level way of considering this process as a life cycle, whereas the second two present it more as a series of discrete steps in a workflow. Figure 2 illustrates a simplified view of data from a researcher's point of view and Figure 3 is the life cycle from a curator's perspective. The steps in Figure 4 are applicable to a researcher who performs processing on their data and Figure 5 is another variation (Humphrey, 2006).

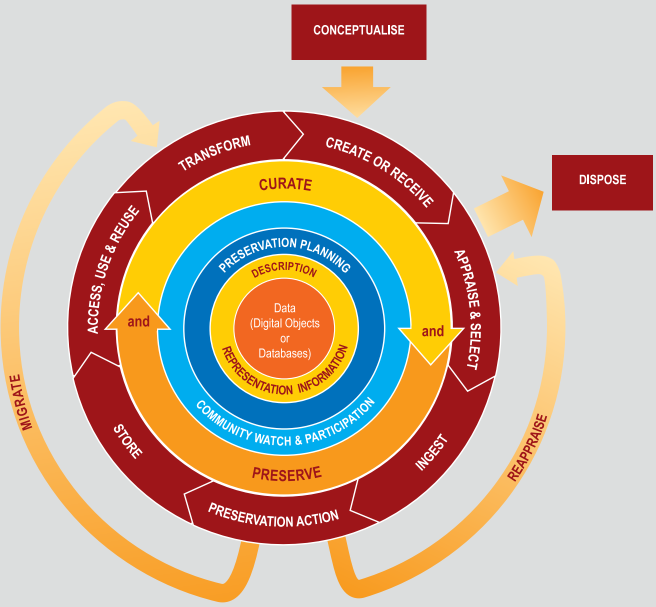
Figure 2: Data life cycle as a user/creator of data



All of these variations can be represented by the following three tasks:

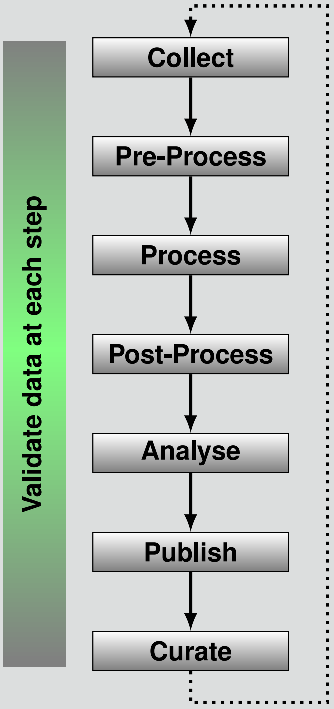
- Obtaining the data
- Using the data
- Looking after the data

Figure 3: Data life cycle from the perspective of a curator (Digital Curation Centre, 2016)



This version of the data life cycle is defined by the Digital Curation Centre and is relevant to a curator of data.

Figure 4: Stages in data life cycle as a user/creator of data, applicable to a researcher who performs processing on their data



First, the data to be processed is **collected**. It is then **pre-processed** to prepare it for the processing step, e.g. cleaning it, editing headers, deleting data fields, and merging with other data. Configuration of the software also happens in the pre-processing step, before starting the **processing**.

Post-processing is where the results are manipulated to extract pertinent data.

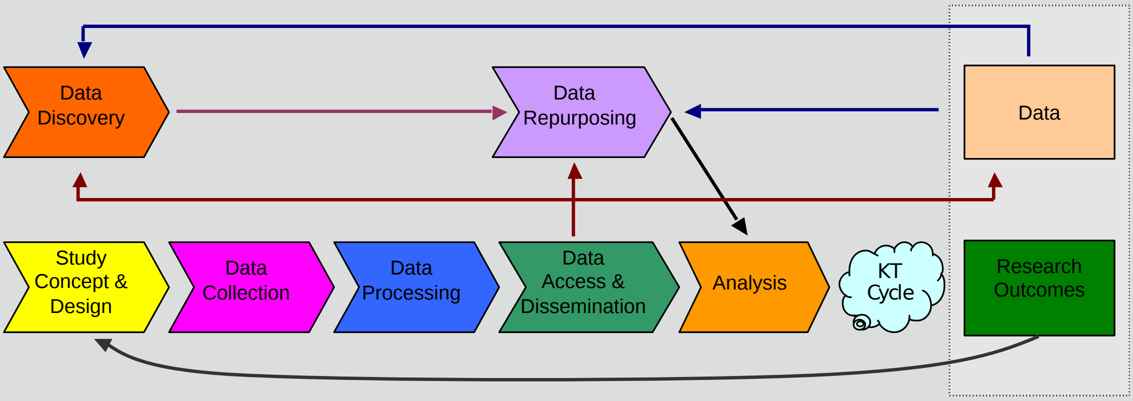
Then the data can be **analysed** qualitatively to assess the value of the produced data perhaps leading to another processing stage, and quantitatively to discover the interesting features of the data which may lead to **publication** if something new or useful is discovered.

The **curation** step involves looking after the data for as long as it is required and then destroying it in an appropriate way – possibly securely. This will depend on the significance of the data, whether it has been used to support publications, and the nature of the data collected, e.g. sensitive or commercial. It is expected that **validation** of data occurs at every step.

Not all stages will be relevant in all cases. For example, some people may not perform any processing steps on their data, just collecting and analysing it, and in other situations data may never be published. It is also expected that there may be multiple iterations within this life cycle where, for example, the post-processing causes a researcher to go back and collect more data.

Figure 5: Stages in data life cycle at a research project level (Humphrey, 2006)

This is another variation, showing the earlier step of 'Study Concept & Design'. Planning for data collection is an essential part of the life cycle. In the figure, 'KT Cycle' is an abbreviation for 'Knowledge Transfer Cycle' which includes publishing, conferences, books and reports.



Part II

Case Studies

This table illustrates how the categories introduced in Part I are used in the case studies that follow. It is not comprehensive as all researchers have their own methods.

● **Case study provides good example**
○ **Also relevant in case study**

	Human Genetics	Fatigue Test	CFD	Crystallography	Archaeology
Sources of data					
Scientific experiments	○	●		○	
Models or simulations	○		●		
Observations		○			●
Derived data	○	○		●	
Reference data	●	○			○
Types of research data					
Electronic text documents	○		●	○	●
Spreadsheets	○	●			●
Laboratory notebooks, diaries					●
Questionnaires, transcripts, codebooks					
Audiotapes, videotapes					●
Photographs, films					●
Examination results					
Specimens, samples, artefacts, slides	●			●	●
Digital objects	○	○	●	○	●
Database schemas					○
Database contents					●
Models, algorithms, scripts	○	○	●	○	
Software configuration			●		
Software pre-process files	○	○	●	○	
Software post-process files	○	○	●	○	
Methodologies, workflows, procedures					●
Experimental results				●	
Metadata					●
Other data files					
Electronic representation of data					
Textual	●		●		●
Numerical	○	●			●
Multimedia		○	○	○	●
Structured	●			●	●
Software code			○		
Software specific			●	○	●
Discipline specific	●			○	●
Instrument specific				●	●

1 Medicine: Human Genetics

This chapter discusses an example of data produced in medical research. This provides a good illustration of using reference data in research, in that the sample data is compared against a reference data set.

Data categories in this case study

● **Case study provides good example**
○ **Also relevant in case study**

Sources of data		
Scientific experiments	○	Analysis of gene sequence
Models or simulations	○	Genome Analysis Toolkit processing
Derived data	○	Aligned data
Reference data	●	Human genome reference sequence
Types of research data		
Electronic text documents	○	Journal publication containing details of discoveries
Spreadsheets	○	For gene sequence analysis
Specimens, samples, artefacts, slides	●	The DNA sequence
Digital objects	○	Gene sequence figures
Models, algorithms, scripts	○	Genome Analysis Toolkit files
Software pre-process files	○	FASTQ file
Software post-process files	○	Novoalign output
Electronic representation of data		
Textual	○	Journal publication containing details of discoveries
Numerical	○	Spreadsheets containing analysis
Structured	●	FASTQ files (structured, text-based format)
Discipline specific	●	FASTQ files

Data life cycle steps in this case study

Data life cycle stages		
Collect	●	Analysis of DNA sequence using sequencing machine
Pre-Process	●	Align sequences against reference genome sequence with *Novoalign*
Process	●	Process data with *Genome Analysis Toolkit*
Post-Process	●	Filter results with *SIFT*
Analyse	●	Analyse results with Microsoft Excel
Publish	●	Discovery of genetic cause of a disease to a journal
Curate	●	Upload sequence data to public genome databases

Background

Obtaining the data

Researchers into human genetics take a DNA sample and analyse it using a sequencing machine which produces short sequence *reads* representing the sequence as millions of fragments. The fragments represent the sequence of 3 billion nucleotides producing a dataset of up to 50 GB. The most cost-effective strategy at this time is to only sequence the protein coding regions (exome) which is about 1% of this data. The data generated is in a text-based format known as *FASTQ*.

Using the data

The FASTQ data is pre-processed by a software package called *Novoalign* to align the short reads against a complete human genome reference sequence. The aligned data can then be processed using the software tools in the *Genome Analysis Toolkit* to identify regions that are different from the reference data set.

Comparing sequences from two subjects' samples will identify thousands of differences, the majority of which make no difference to the protein that the gene codes for. *Synonymous* differences do not change the protein, but *non-synonymous* differences will. A DNA sequence with a *frameshift mutation* codes for a protein that is considered to be non-functional. However, an average adult will have a large number of these differences and still be healthy and research has only just begun to understand the variants and how they can cause diseases.

Filtering against genome databases permits known diseases to be identified by comparing the variants identified with those that are known causes of disease, and the effect a variant may have on the protein can be calculated by tools such as *SIFT (Sorting Intolerant From Tolerant)*.

The variant data produced is in tabular form with one row per variant and includes:

- The location of the variant in the protein
- The values of the reference nucleotide and the variant
- The quality of the reading
- How many times the section of protein has been examined and found to be different (because each part of DNA is analysed multiple times)
- Whether the variant has been seen before

Microsoft Excel is frequently used for analysis of the variant data.

Looking after the data

Data is stored in a number of formats as it is converted from type to type for each stage. New discoveries are fed back to the community by uploading to sequence databases so tools like SIFT can take advantage of the new data.

Figure 6: Viewing genetic sequence mutation data (Broad Institute, 2016)

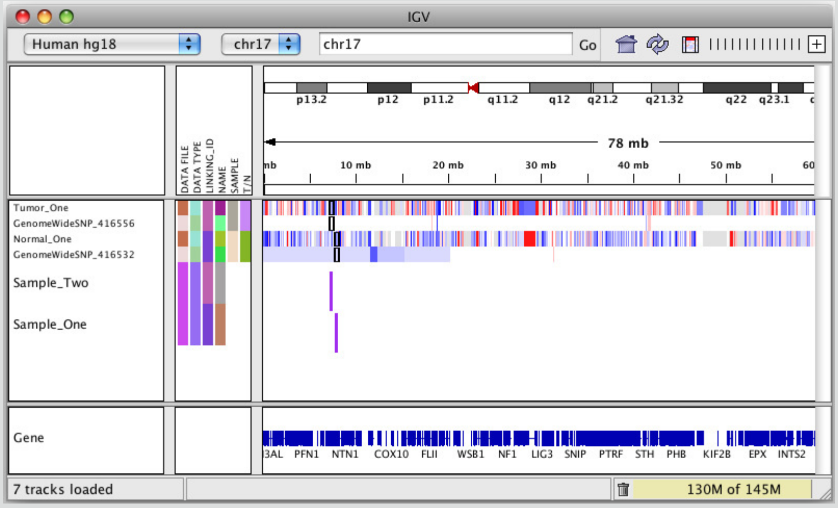
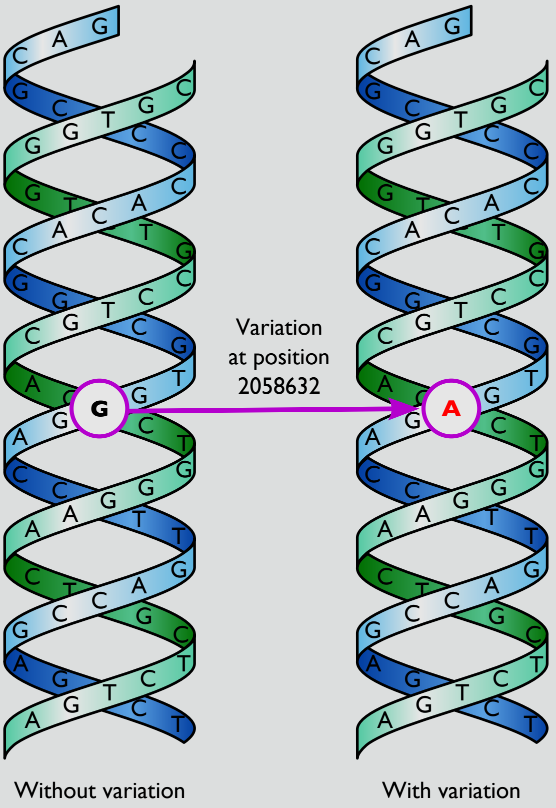


Figure 7: Variation in the *TSC2* gene, identified as one possible cause for *tuberous sclerosis*, a genetic disorder that causes tumors in organs around the body



Summary

In this case study, the researcher collects DNA sequence data and, after preparing the data by aligning it, processes it to find regions that differ from a reference data set. Analysis of the data can involve comparing it against genome databases and manipulation in spreadsheet software. Publication will occur when patterns that relate to diseases are identified, which will then help improve the existing genome databases and analysis tools.

This case study provided good examples of a wide range of research data including:

- Using reference data (the complete genome reference sequence, and the genome databases);
- Discipline specific data (the FASTQ files containing the DNA sequence reads).

2 Materials Engineering: Long crack growth fatigue testing

This section will be used to give an example of data produced in materials engineering. This gives a good illustration of acquiring data through scientific experiment and then performing some processing on it.

Data categories in this case study

● Case study provides good example

○ Also relevant in case study

Sources of data		
Scientific experiments	●	The fatigue test
Observations	○	Monitoring PD values to find threshold
Derived data	○	Smoothed PD readings in Excel
Reference data	○	K values for specimen geometry and load combination; PD calibration values for the material.
Types of research data		
Spreadsheets	●	Manipulated PD readings in Excel
Digital objects	○	Graph of da/dN versus ΔK; Images of the material's microstructure.
Models, algorithms, scripts	○	Smoothing algorithm
Software pre-process files	○	Raw CSV data of PD readings
Software post-process files	○	Manipulated PD readings in Excel
Electronic representation of data		
Numerical	●	CSV and Excel files
Multimedia	○	Graph of da/dN versus ΔK (vector file); Material's microstructure (bitmap file).

Data life cycle steps in this case study

Data life cycle stages		
Collect	●	PD readings from fatigue test
Pre-Process		
Process		
Post-Process	●	Manipulating/smoothing PD readings
Analyse	●	Plotting of stress against number of cycles to failure (S-N curve)
Publish	●	Journal paper containing properties of a material
Curate	●	Upload data to materials data repository

Background

Obtaining the data

Starting with a sample of material that contains a crack, the growth of the crack is monitored as loads are applied to it. An electric current is passed through the sample and fluctuations in resistance or impedance – depending on whether direct or alternating current is being used – are detected by measuring the change in potential difference (PD) across the sample. PD increases as the resistance or impedance of the material increases, caused by the growth of the crack.

K is a theoretical value representing the stress intensity of a crack based on the applied load and the crack's size and geometry. The range of *K* values to be applied to a sample in a fatigue test, known as *ΔK*, is calculated in advance and load is adjusted to match the target *ΔK* as the length of the crack changes. *K* values for a specimen geometry and load combination are usually pre-existing, having previously been calculated. If they are not available, finite element analysis methods are used to calculate them.

Using the data

The raw data generated from a fatigue test consists of the PD readings in relation to time. The length of the crack (*a*) is calculated from the PD voltage data and the number of loading cycles (*N*) is known from the time elapsed. The rate at which the crack grows with respect to the number of loading cycles can then be calculated (*da*/*dN*) and compared to the range of stress intensity (*ΔK*) to give a measurement of crack driving force and the material's ability to resist crack propagation at different stress intensities.

Calculating the length of the crack from a given PD value is possible because of previous calibrations with the material and specimen geometry. When using a material and specimen geometry where no data is available, early experiments do not necessarily yield any useful data other than data that can be used for PD calibration. It is helpful to find someone else who has already performed fatigue tests on a particular material and can give starting values for calculating crack length.

The captured data is often quite noisy so it may be necessary to smooth it during analysis. There are many methods for this, including skipping obviously erroneous sampled values or using a best-fit curve.

Looking after the data

Raw data is stored as CSV and its manipulation is usually done using spreadsheet software such as Microsoft Excel.

Figure 8: Fatigue Test Configuration

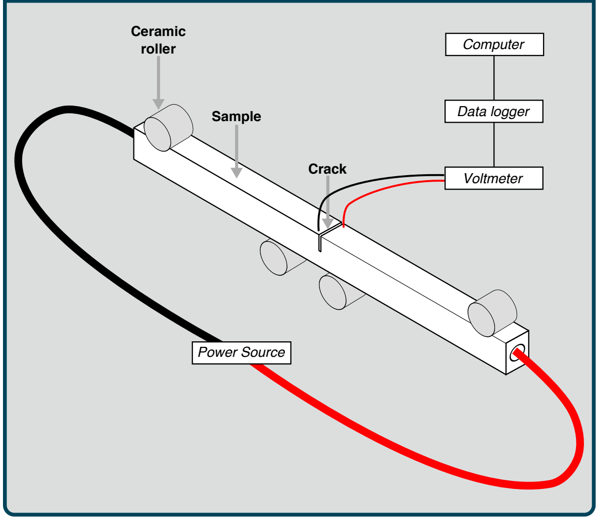
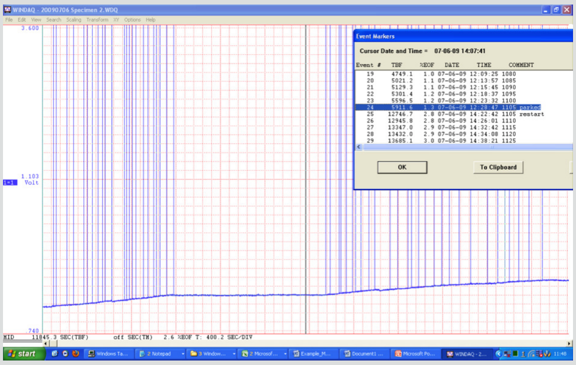


Figure 9: Collecting data during fatigue test

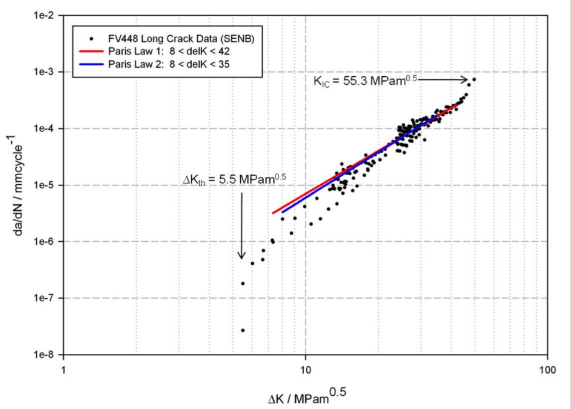


```
Cursor Date and Time =    07-06-09 14:07:41

Event #    TBF      %EOF   DATE      TIME     COMMENT
   19      4749.1    1.0 07-06-09 12:09:25 1080
   20      5021.2    1.1 07-06-09 12:13:57 1085
   21      5129.3    1.1 07-06-09 12:15:45 1090
   22      5301.4    1.2 07-06-09 12:18:37 1095
   23      5596.5    1.2 07-06-09 12:23:32 1100
   24      5911.6    1.3 07-06-09 12:28:47 1105 parked
   25     12746.7    2.8 07-06-09 14:22:42 1105 restart
   26     12945.8    2.8 07-06-09 14:26:01 1110
   27     13347.0    2.9 07-06-09 14:32:42 1115
   28     13432.0    2.9 07-06-09 14:34:08 1120
   29     13685.1    3.0 07-06-09 14:38:21 1125
```

```
MID     11845.3 SEC(TBF)      off SEC(TM)    2.6 %EOF T:  400.2 SEC/DIV
```

Figure 10: *da/dN* versus *ΔK* plot



Summary

In this case study, the researcher collects potential different readings from a voltmeter whilst applying loads to a sample of material containing a crack. Using this data, along with calibration data, it is possible to know how quickly a crack grows in a material at different levels of applied stress.

This case study provided good examples of a wide range of research data including:

- Collecting data through scientific experiment (the fatigue test);
- Using spreadsheet software to manipulate and refine CSV data by smoothing it;
- How transforming the data is useful for analysing raw data (in this case by generating a plot of stress against number of cycles).

3 Aerodynamics: Simulations Using Computational Fluid Dynamics

This section gives an example of data usage in aerospace engineering. This gives a good illustration of using models or simulations to acquire data.

Data categories in this case study

● **Case study provides good example**
○ **Also relevant in case study**

Sources of data		
Models or simulations	●	Air flow simulation in Fluent
Derived data	○	Data produced from simulation
Reference data	○	Properties of air in simulation
Types of research data		
Electronic text documents	●	Text document describing the simulation
Digital objects	●	Images produced from TecPlot; animations of air flow
Models, algorithms, scripts	●	Meshes; geometries; MATLAB scripts
Software configuration	●	Fluent case files
Software pre-process files	●	Meshes; geometries
Software post-process files	●	Fluent output
Electronic representation of data		
Textual	●	Text document describing the simulation
Multimedia	○	Images produced from TecPlot; animations of air flow
Software code	○	MATLAB scripts; Fluent functions (UDFs) in C
Software specific	●	Meshes; geometries

Data life cycle steps in this case study

Data life cycle stages		
Collect	●	(Collection of data is through the simulation in the following stages)
Pre-Process	●	Creation of meshes and geometries
Process	●	Fluent simulation
Post-Process	●	TecPlot analysis
Analyse	●	MATLAB turbulence and vortex analysis
Publish	●	The findings of the analysis, e.g. how to improve rotor design
Curate		

Background

In aerodynamics, computational models are used to understand air flow, for example, around an aeroplane wing or through an air conditioning unit. Simulation of the air flow is known as *Computational Fluid Dynamics* (or *CFD*) and might be used for improving the performance of the wing or efficiency of the air conditioner. This case study details the process of a *rotor wake simulation* which involves modelling the flow of air around a helicopter rotor operating close to the ground.

Obtaining the data

Software known as *ANSYS Fluent* takes a number of data items in order to simulate the flow of a fluid (in this case the air affected by a helicopter's rotor blade). This includes a *mesh* which describes the volume of air being simulated broken up into a 3D grid of fluid volumes and a *geometry* to describe the shape of the objects in the simulation. In this case study, a method known as the *actuator line method* allows the geometry data to be substituted for formulae that calculate how the rotor blade interacts with the air by producing lift.

The mesh and some *user-defined functions* (*UDF*) written in C that handle the actuator line method calculations are loaded into Fluent. A Fluent *case file* is created describing the model's and software's configuration and then computation can begin. Usually this occurs on a *high performance computing* cluster of machines. Using 16 CPU cores, simulating 1 second of air flow takes about 60 hours, broken down into time steps of 0.0005 seconds. A data file is saved every 5 time steps representing 0.0025 seconds of simulation. The total data generated is around 300 GB per 1 second of simulated flow.

Using the data

Once the simulation is complete, the data can be used for a number of purposes. Fluent can be used to produce images of the data showing flow development and the mesh quality to assess whether the simulation needs to be rerun with a different configuration or mesh. Software known as *TecPlot* can be used to produce pictures of the processed data either to help identify if the flow is developed and whether to continue the simulation, or to identify the points of interest in the simulation for validation. This qualitative analysis helps to improve the simulation. A program known as *MATLAB* is used to explore the data in a quantitative way for more demanding numerical operations such as turbulence and vortex analysis.

Looking after the data

Aerodynamicists have a clear workflow they follow, producing data at each stage. When investigating an extended time period of simulated air flow the researcher may produce a lot of data, which needs looking after. Depending on how long the simulation takes to generate the data, the approach taken by the researcher is different. A simulation taking many months to run produces data that needs more care than one that takes a few days, but in both cases, it is the code that is used to produce the simulation that is the most critical.

Figure 11: The Helical Air Flow Produced By A Helicopter Rotor In TecPlot

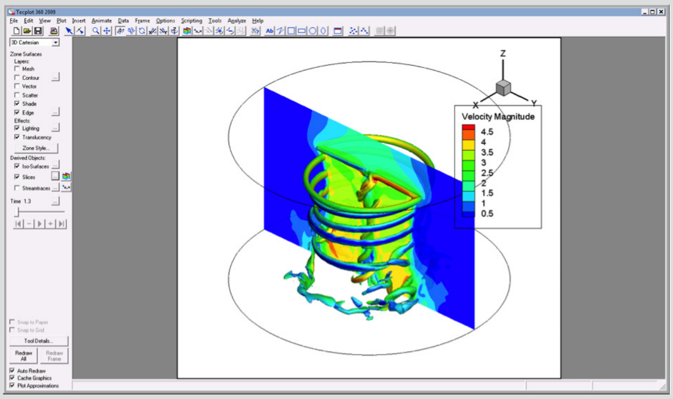
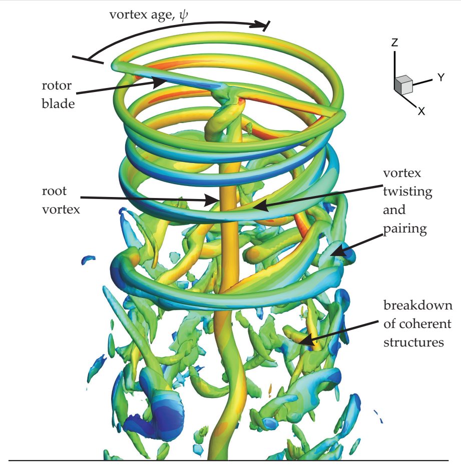


Figure 12: The Air Flow After Analysis As It Might Appear in Published Work



Summary

The researcher in this case study produces data by creating a mesh and some user-defined functions, and then using software to simulate air flow. The results are then analysed by producing images and videos of the air flow to identify points of interest or assess whether the simulation was sufficient. The results can also be explored with other software to find features such as turbulence and vortices.

This case study provided good examples of a wide range of research data including:

- Collecting data using models or simulations (the Fluent job executed on an HPC cluster);
- Digital objects as data (the images and videos representing the air flow);
- Models, algorithms and scripts (the meshes, geometries, UDFs as C code, and MATLAB scripts);
- Software configuration files (case files for Fluent);
- Software pre-process files (meshes and geometries also come under this category);
- Software post-process files (MATLAB files);
- Software specific data (meshes and geometries are specific to Fluent).

4 Chemistry: Crystal Structures

We will use manipulating X-ray data to analyse crystal structures as an example for chemistry. This provides a good illustration of deriving data. In this case, taking data from a scientific experiment and correcting and refining to to extract only the useful data.

Data categories in this case study

● Case study provides good example
○ Also relevant in case study

Sources of data		
Scientific experiments	○	The X-ray examination of the crystal
Derived data	●	The extraction of h, k and l Miller indices

Types of research data		
Electronic text documents	●	Detail regarding properties of sample
Specimens, samples, artefacts, slides	●	The crystalline sample
Digital objects	○	Crystal structure images and videos
Software pre-process files	○	Raw X-ray data
Software post-process files	○	`.hkl` data
Experimental results	●	Raw X-ray data from diffractometer

Electronic representation of data		
Multimedia	○	Crystal struture images
Structured	●	`.hkl` structured text data
Software specific	○	*Rigaku*'s CrystalClear data files
Discipline specific	○	`.hkl` data, CIF files
Instrument specific	○	*Rigaku* diffractometer data

Data life cycle steps in this case study

Data life cycle stages		
Collect	●	The X-ray examination of the crystal
Pre-Process		
Process		
Post-Process	●	The extraction of h, k and l Miller indices
Analyse	●	Iterative process to find a model that matches the sample
Publish	●	Submit to journal new chemical or new form of known chemical
Curate	●	Upload to Crystallographic Data Centre or eCrystals web site

Background

Obtaining the data

In chemistry, X-rays are used to examine crystalline samples in order to determine the properties of a newly-created chemical or to analyse the structure of a known chemical. Along with the X-ray data, the history of how the sample was created is required to assist with the analysis.

An X-ray diffractometer is used to perform the scan, in this case manufactured by *Rigaku*. The diffractometer is driven using software supplied by the manufacturer, for example, Rigaku's *CrystalClear*. The instrument produces raw image data made up of 300–400 segments representing a sphere. Each segment's data is stored in a software-specific file of around 4 MB in size, which collectively gives a data set of approximately 1 GB.

The CrystalClear software is used to perform corrections on the raw data and execute algorithms to extract the useful data, producing a single structured text file less than 100 MB per data set. This file contains h, k and l Miller indices of the crystal planes and data relating to spots in the image (known as σ and F^2). It is this file that is used for the remaining investigation on the researcher's own workstation outside the laboratory.

Using the data

This `.hkl` file can then be used to determine the sample's properties. This is done by producing a model which is assumed to match the characteristics of the sample from which another `.hkl` file is produced and compared to the sample's. This process of creating a model, producing a comparison `.hkl` file from the model and then verifying it against the sample's data may be performed many times until the correct model is identified.

The model is then saved in *CIF* format (Crystallographic Information File) which is a discipline specific file format – only a few kB in size – used to store the model, as well as other related data such as the experiment's metadata and even textual data for use in publications.

Looking after the data

The raw data produced by the X-ray diffractometer is kept indefinitely and is stored on a server in the Chemistry department, mirrored to a secondary server which is then backed up to tape. This raw data may be used again in the future for producing alternative `.hkl` files if the original file proves insufficient in some way.

The `.hkl`, `.cif` and other files are not stored on the file server as these files are used by the researcher on their own personal computers. It is standard practice to use the experiment ID in the names of these files to ensure they can be traced back to the raw data. Other than that, each researcher is responsible for managing their own working files.

When new chemicals – or a new form of a known chemical – are identified these may be submitted as a paper in a journal. Many journals also now expect the `.cif` file. The `.cif` file may also be uploaded to a data centre such as the *Cambridge Crystallographic Data Centre* as a central repository, or shared via a local data repository such as on the *eCrystals* web site.

Figure 13: A graphical representation of a crystal structure

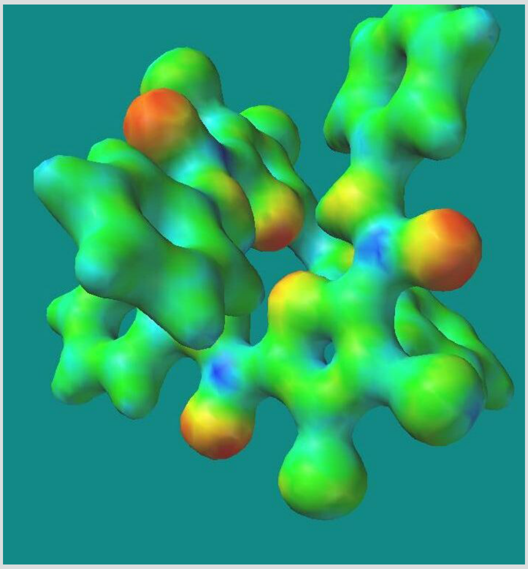
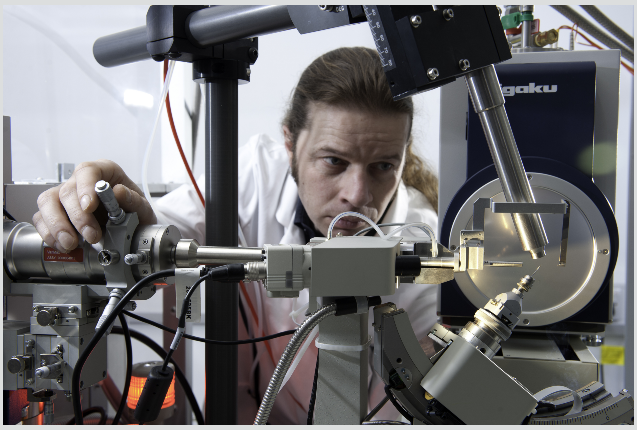


Figure 14: An X-ray diffractometer in use



Summary

The researcher in this case study produces data relating to a crystalline sample using an X-ray diffractometer. The data is then refined by performing corrections and, using algorithms for analysis, produce the pertinent data needed in the next stage (finding a model that best describes the sample).

This case study provided good examples of a wide range of research data including:

- Collecting data by deriving it from some other form (taking the raw data produced by the X-ray diffractometer and processing it to produce a `.hkl` file;
- Collecting data from scientific experiment (X-ray diffractometer);
- Using electronic text documents in research (in this case, to describe how the sample was created);
- Specimens and samples (the crystalline specimen being analysed);
- Experiment results (the data produced by the X-ray diffractometer);
- Structured data (`.hkl` file);
- Software, discipline and instrument specific data.

5 Archaeology: An Excavation

We will use the data collected by archaeologist during an excavation in this example. This provides a good illustration of observational data as well as reference data.

Data categories in this case study

● **Case study provides good example**
○ **Also relevant in case study**

Sources of data		
Observations	●	Details and features about sites and discoveries
Reference data	○	Maps of an area; records of previous work on a site
Types of research data		
Electronic text documents	●	Excavation diary
Spreadsheets	●	Spreadsheets detailing finds, e.g. dimensions and weight
Laboratory notebooks, diaries	●	Excavation diary
Audiotapes, videotapes	●	Excavation site video
Photographs, films	●	Photographs of site
Specimens, samples, artefacts, slides	●	Discoveries from site
Digital objects	○	Digital photogrammetry
Database schemas	○	Excavation details database
Database contents	●	Excavation details database
Methodologies, workflows, procedures	●	Excavation procedures
Metadata	●	IPTC photographic metadata
Electronic representation of data		
Textual	●	Excavation diary
Numerical	●	Spreadsheets detailing finds, e.g. dimensions and weight
Multimedia	●	Photogrammetry; scene visualisations
Structured	●	Excavation database
Software specific	●	*ArcGIS* files
Discipline specific	●	*ARK* (Archaeological Recording Kit) files
Instrument specific	●	Polygon Workbench for driving laser scanner

Data life cycle steps in this case study

Data life cycle stages		
Collect	●	Taking measurements, photographs and other data created during excavation
Pre-Process		
Process		
Post-Process		
Analyse	●	Assessing collected data to verify nothing was missed; looking for patterns in discovered objects
Publish	●	Publication of discoveries
Curate	●	Uploading to the *Archaeology Data Service*

Background

Archaeologists on the Portus Project (University of Southampton, 2016) aim to discover more about the history of Portus, construction of which was begun by Emperor Claudius around 2000 years ago and was the primary port for the city of Rome during imperial times. This case study looks at some of the data collected by an archaeologist on such a project and how it is looked after.

Obtaining the data

Archaeologists employ many modern techniques such as laser scanning and X-ray computed tomography (CT) as well as traditional approaches such excavation and maintaining a diary.

One of the most important forms of archaeological data comes from observations taken before and during an excavation. This is because the nature of archaeology is destructive and, without good quality records, valuable information could be lost. Not only are the observations important, but the way in which the observations were made are also documented. These records help researchers understand the complete story from discovery to publication and to identify anything missed during a dig.

The accuracy of the observational data affects its usefulness, so researchers try to find ways of being as accurate and detailed as possible, for example by using very precise instrumentation for measuring and scanning. A global positioning system (GPS) or a *total station* containing a theodolite, data logger and distance meter helps with surveying an area, and *digital photogrammetry* is used for recording site layouts in detail.

Descriptions of items of interest, such as the colour and properties of an object, also require precision. For example, to describe colours the *Munsell Book of Color* is used which contains hundreds of reference colours palettes with associated codes. To describe what an object is, archaeologists use a *typology* to categorise it and a thesaurus such as the *Getty Art & Architecture Thesaurus* containing a controlled vocabulary for describing it. Other resources include *INSCRIPTION* provided by the Forum on Information Standards in Heritage containing recommended wordlists for describing sites.

In addition to observational data and the associated documentation, archaeologists use a lot of reference data, especially when planning a dig. This comes from a number of sources including historical records, maps, geophysical data, previous work done at a site and even the researcher's earlier work.

Recording of archaeological data is done using specialist systems such as the *Archaeological Recording Kit* (ARK) which uses a web front-end to populate a database. Another tool is a spreadsheet for recording details of discoveries. Photographs are tagged with the name of the photographer, project, GPS data and other appropriate information using the IPTC metadata standard.

Using the data

Archaeologists tend to work in teams responsible for different parts of a dig such as surveying, excavating and the team responsible for recording finds. Data collected by one team must be easily accessible and understandable to others to ensure mistakes are not made and important details are not missed.

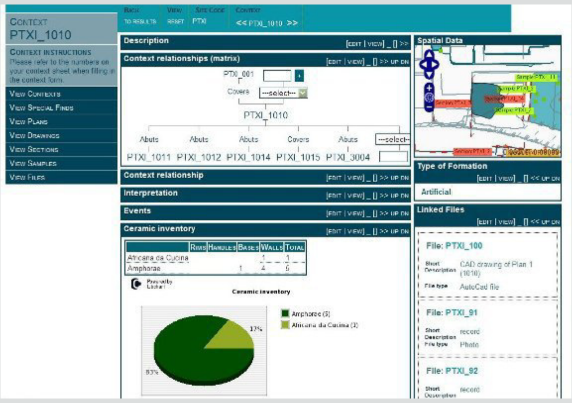
At the end of a season the excavation is written up. This gives a high level view of the work and helps missing data to be identified as soon as possible. Some researchers make very specific use of the data, for example taking the details of the ceramic objects and analysing their properties, perhaps looking at the relative sizes of all bowls across a site, when they were excavated and where. At a much higher level, some researchers may take data from a number of sites across a country and look for patterns in their distribution.

Looking after the data

In order to achieve this level of sharing and thoroughness, archaeologists use specialist software and have strict guidance and standards in place to ensure nothing is missed at any step.

The *Archaeology Data Service* is used as an open-access repository for hosting data. It also provides guidance on how to look after data as well as a place for dissemination of research (University of York, 2016).

Figure 15: The ARK software in use



Context instructions
Please refer to the numbers on your context sheet when filling in the context form.

	Rims	Handles	Bases	Walls	Total
Africana da Cucina				1	1
Amphorae			1	4	5

Artificial

File: PTXI_100
Short Description: CAD drawing of Plan 1 (1010)
File type: AutoCad file

File: PTXI_91
Short Description: record
File type: Photo

File: PTXI_92
Short Description: record

Figure 16: Computer generated simulation of interior of building at Portus

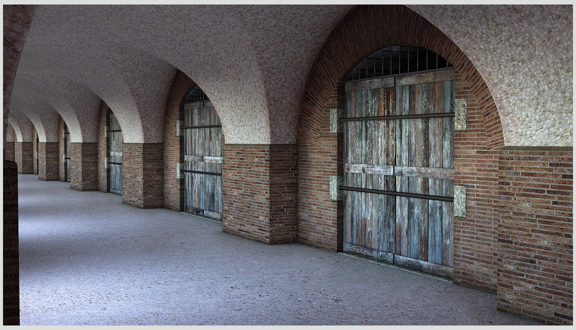


Figure 17: Laser scanning the Cistern complex at Portus using a Leica ScanStation

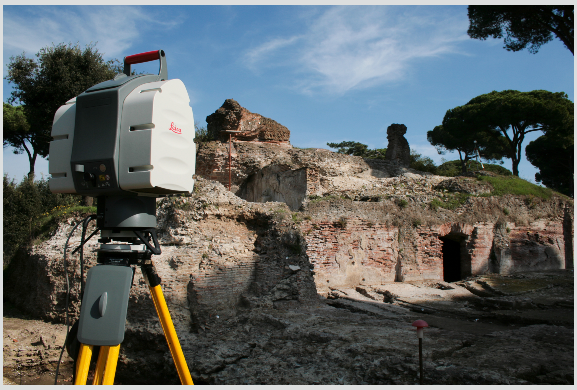
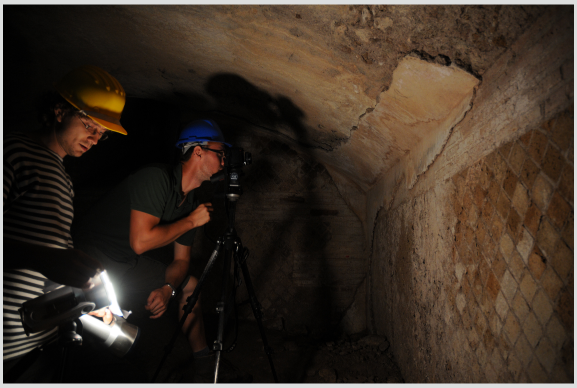


Figure 18: Taking near infrared photos in Imperial Palace



Summary

The researcher in this case study produces data relating to an archaeological excavation. The data comes from a number of sources such as a total station, digital photogrammetry, an excavation diary and previous work.

This case study has provided some good examples of the following research data:

- Creating research data by observation (by accurately recording as much detail as possible so data is not lost while excavation takes place)
- Using reference data (by using data from previous digs as well as geophysical and geographical data when planning a dig)
- Using procedures as part of research (archaeologists follow a set of guidelines on how to perform an excavation. If they didn't, mistakes might happen and data might be lost in the process of the dig)
- Specimens, samples and artefacts (found during the excavation)
- Using a database as part of research (to record excavation details)
- Using metadata (metadata is attached to photographs to improve their usefulness)
- Software, discipline and instrument specific data

Part III
Data Management Practices

1 Storing data

With all the data that will be collected as part of your research it is critical that the data is managed properly. This section lists methods that will help you find your data again and discusses data formats, an understanding of which will help with long-term preservation.

Finding the data again

Methods that can help with finding the data again include:

- Follow sensible file naming conventions.

  Try to use names that match your environment and contain:

  1. Something meaningful to you (such as what you are doing with the file)
  2. Something meaningful to someone else (such as an experiment number or project name)

  Examples of file naming

  Good

  materialN18_sample9_smoothed.csv

  20150115a_projectabyz_pressuresensor2.dat

  Bad

  smoothed.csv

  file1.txt

  When using dates in file names, format them as year, month, day (yyyymmdd) to help with sorting file listings, e.g. 20150525

- Tagging and search (can you remember the tag later?)
- Using a database or spreadsheet to track data – or in your logbook
- Structuring folders on the disk (one big-flat folder with hundreds of folders versus hierarchical tree)
- Storing it online using a discipline specific repository or archival system (local versus open access)
- Storing it in an institutional repository or a general repository (e.g. Figshare, Mendeley Data)
- Link to a publication or suitable write-up if there is one, to help others understand the data

Data formats

Your research data may be used by someone else in the future, possibly even you if you need to refer back to it. It has already become mandatory that some research data must be held for ten years after the last time it was accessed with the expectation that it will remain usable throughout that time. What if the software is not available any more when the data is needed? To prepare for this eventuality, the format used to store the data needs to be considered.

- Save or export data to an ASCII text format when possible – one of a number a specialist data formats for encoding text data which is understood by many of the tools provided with a computer's operating system. Data stored in ASCII text files is likely to still be understandable even if the software that generated the file's structure format is no longer available. Structured text file formats such as CSV and XML can also be saved as ASCII. Microsoft Word and other word processing software can even export to HTML (an ASCII format) which can allow some formatting to be retained and provide some protection if the software is no longer available.

  The most current text encoding standard is known as The Unicode Standard. The Unicode UTF-8 format is the closest to ASCII as the first 127 characters of the character set are the same, providing a degree of compatibility with 7-bit ASCII (meaning ASCII is valid UTF-8). UTF-8 is now commonly used for HTML and XML files.

- When storing data in other formats, using formats with openly published specifications provides some protection. Even if the software is no longer available, the specifications might permit critical data to be retrieved. For example, the OpenDocument (produced by OpenOffice and LibreOffice as well as others) or Office Open XML (produced by Microsoft Office) formats are good choices for storing a word processed document or spreadsheet.

- For documentation, apart from the OpenDocument file formats mentioned previously, PDF could be considered, especially PDF/A which is a standardised version of Portable Document Format that is better suited for long-term archival.

- Figures can be stored in a vector format, which is made up of lines and paths, or a raster format, which is made up of dots (e.g. a photo). Vector formats such as SVG and EPS are often a better choice for illustrations than raster formats such as JPEG and PNG due to loss of quality incurred when scaling raster images. SVG and PNG are open standards which may make them better suited to long-term archival.

- For audio, consider using a lossless format such as Xiph.Org's FLAC (which is also an open format and royalty-free) instead of a lossy format like MP3 which reduces the size of an audio file by removing information – not ideal for data preservation purposes. To avoid patent licencing issues associated with the lossy MP3 standard, consider Xiph.Org's Vorbis or Opus.

- For video, it is wise to check the format of the video to ensure the format is suitable for its purpose and data retention requirements. If the video is uncompressed, there will be a trade-off to make about whether to leave it like that to keep the original data for data preservation purposes or to encode it into a compressed format to reduce the size of the file. Video codecs require careful choice depending on the use of the video, and a decision also has to be made about what container format to use to combine the video and audio components. Container formats include Matroska, Ogg, AVI and MP4. Matroska and Ogg are open standard, free formats, with Matroska capable of containing virtually any format.

  If the video is not being created for broadcast on television or for paid use on the internet, one of the MPEG codecs would be appropriate, with an MPEG container (e.g. MP4). If patent licencing is a concern, then it might be worth sticking with one of the Google codecs such as VP8 or VP9, or Xiph.Org's codecs such as Theora or Daala; the Matroska container will work great with these.

Can you still access your data in 20 years?

File format recommendations

File format	Extension
Textual data	
ASCII text	`.txt`, etc.
OpenOffice – OpenDocument Text	`.odt`
MS Office – Open XML	`.docx`
PDF/A (a standardised version of PDF)	`.pdf`
HTML	`.html`, etc.
Numerical data	
Structured text	`.csv`, `.xml`, etc.
OpenOffice – OpenDocument Spreadsheet	`.ods`
MS Office – Office Open XML	`.xlsx`
Multimedia	
Vector image: Scalable Vector Graphics (SVG)	`.svg`
Raster image: Portable Network Graphics (PNG)	`.png`
Audio – royalty free: FLAC (lossless), Apple Lossless (ALAC), Vorbis (lossy), Opus (lossy)	`.flac`, `.alac`, `.ogg`, `.opus`
Audio – patented: MP3 (lossy), Windows Media Audio (WMA) (lossy/lossless, not open)	`.mp3`, `.wma`
Video codec – royalty free: VP8/VP9 (SD), VP8/VP9/Dirac (HD), VP10/Dirac/Daala (UHD)	See containers
Video codec – patented: Xvid (SD), x264 (HD), x265 (UHD)	See containers
Video container – royalty free: Matroska, Ogg	`.mkv`, `.ogv`
Video container – patented: MP4, AVI	`.mp4`, `.avi`

More about audio and video

AVI, MP3 and the MPEG video codecs (H.264, H.265, etc.) are arguably more supported and are good choice, but licencing and patents may limit their ultimate usefulness.

Open standards provide long-term protection but one additional consideration with video (and audio) is whether there are any patents which require the payment of a licence when using the codecs. For example, the open standard encoder x264 can be used to encode videos for the open standard H.264/MPEG-4 AVC – used on Blu-ray discs and by the YouTube web site among others – but this standard is protected by many patents which the MPEG LA firm licences. No licence is required if the video is broadcast via the internet for free, but other uses may involve fees.

For ultra-high definition videos, HEVC/H.265 is the successor to H.264 and x265 is the equivalent open source encoder.

For lower resolution video (DVD quality) the FFmpeg tools allow encoding into the H.262/MPEG-2 Part 2 standard used by DVD players, but a smaller file size with the same resolution can be achieved with the later MPEG-4 Part 2 standard using an open source encoder such as Xvid.

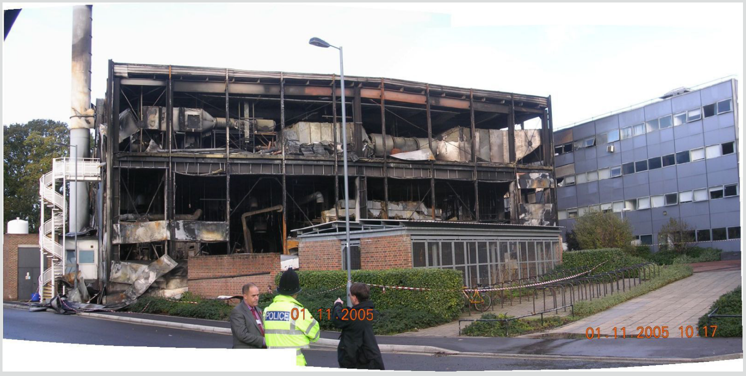
Several open standard, royalty-free codecs do exist but are not as common and are therefore not supported by software as well. The best current contenders are On2/Google's VP8 and Google's VP9. The WebM container supported by the Firefox, Chrome and Opera browsers is based on the Matroska media container and uses the VP8 or VP8 codec along with Opus or Vorbis for audio, both open source and royalty-free. Work started this year by the Alliance for Open Media to create a new codec for ultra-high definition video that will be licence-free and open, and will be based on Xiph.Org's Daala, Cicso's Thor and Google's VP10 (all open, royalty-free codecs). One final codec worth mentioning is BBC's open source, royalty-free codec, Dirac, which supports lossless compression for high-definition video.

2 Backups

On 30 October 2005, the University of Southampton's Mountbatten building caught fire and many students and staff were suddenly faced with the possibility of having lost months' worth of research.

Data loss more often happens in much less dramatic circumstances. Most of the time, it's a hard drive failing, equipment such as a laptop being lost or stolen, or simply down to user error, e.g. files being deleted, or vendor error, e.g. a software bug causing corruption. In order to protect your data, you should ensure you make regular backups, **preferably to more than one physical location.**

Figure 19: The Mountbatten Building after the fire (Bullas, 2005)



Scans or Photocopies

Research data is not just in the form of electronic data; log books, slides and documents can also be lost. Do not forget these types of research data and ensure you have some form of copy if they are critical to your research.

Where to backup

University file store:
iSolutions provide file storage on a server, which is backed up so it is recommended that you make use of this. This can be accessed through the My Documents folder on University Windows PCs. You should be aware that backups are only retained for 3 months.

The file store can be accessed through the My Documents folder. When not using a University Windows PC, the file store can be found at the following location, replacing <id> with your username:

Windows:

```
\\filestore.soton.ac.uk\users\<id>
```

Mac/Linux:

```
smb://filestore.soton.ac.uk/users/<id>
```

External hard drive:
Copying your files onto an external hard drive (or another computer) is a simple way to backup your data. The hard drive should not be stored together with the computer (in case of a fire, flood, stolen laptop bag, etc.). It is usually a manual process, although software exists to do it automatically (see panel).
Regularly checking recent backups have been performed is vital.

Useful backup tools

You may find the use of tools such as *rsync* or *SyncToy* helpful to automate your own backups, such as synchronising files between a laptop with files stored locally and a server. Here are some built-in Mac and Windows backup tools:

Operating system	File backup	System backup
Windows 7	Backup and Restore	Backup and Restore
Windows >8	File History	System Image
Mac OS X >10.5	Time Machine	Time Machine

Backup tools

Built-in Windows tools:
On Windows 7, this used to be the *Backup and Restore* tool, and for Window 8 and 10, *File History* is used for file backups and *System Image* to backup the system.

Built-in Mac OS X tools:
For those running Mac OS X 10.5 or above, activate *Time Machine* for an easy, automatic solution to an external hard drive. For those who run virtual machines, it is a good idea to exclude these files from backups as they will quickly fill your hard drive and backing up a file when open may cause corruption. If you require a Time Machine backup of your VMs, please ensure you use snapshots which Time Machine can backup safely and then you will always be able to roll back to the snapshot.

SyncToy:
For those running Windows machines, *SyncToy* is free software available from Microsoft which will help you set up automatic backups of files across drives. It is old but can still prove useful.

rsync:
A file synchronisation and backup command-line tool for Linux and Mac OS X. The following example provides a simple backup command:

```
rsync -avi sourcefolder/ destfolder/
```

robocopy:
A file synchronisation and backup command-line tool for Windows. The following example provides a simple backup command:

```
robocopy sourcefolder destfolder /z /e /v
```

Non-free alternatives:
For extra protection, a service such as CrashPlan or BackBlaze is useful for backups. CrashPlan and BackBlaze charge a few pounds a month – $5.99 for CrashPlan and $5 for BackBlaze for 1 computer – and provide backup software and unlimited backup space on their servers. Consider the time it will take to transfer data back from the service when doing a full restore and check whether they will ship a hard drive with the data which can be quicker than restoring over the internet. See panel below ('Warning!') about storing data with a third party.

Warning!

Be careful when considering storing data with a third party, especially with providers such as CrashPlan, BackBlaze and Dropbox (discussed below) – if your data is sensitive it may not be allowed to leave the University or may need to be protected through encryption. Be particularly careful if the data is stored on servers outside of the United Kingdom. If you are relying on servers outside of the EEA please seek legal advice from Legal Services.

If your project requires ethics approval, storage should always comply with information provided as part of that process.

Finally, it is important to consider what happens when you leave the University. OneDrive for Business (discussed below) gives you 30 days of grace when you leave the University **and then your OneDrive for Business account gets deleted**.

A word on Dropbox and similar services

Dropbox ≠ backup!

Dropbox synchronises deletes as well as modifications!

A full restore can be slow, difficult and messy.

See panel on previous page ('Warning!') about storing data with a third party.

Typical features of Dropbox-like services

Feature	Preferred alternatives
File synchronisation	OneDrive for Business (but be aware of limitations below); peer-to-peer/private cloud, e.g. Resilio Sync (btsync) or Syncthing
Collaborative working	University file store, or OneDrive for Business (but be aware of limitations below)
File sharing	University Dropoff service at `https://dropoff.soton.ac.uk`

It may be tempting to use a service like Dropbox for data storage. Dropbox is for synchronisation of files between computers and sharing of data. A small amount of space is given for free and there are also paid plans. Alternatives include Google Drive and Microsoft OneDrive, and WeTransfer is also popular. Always watch out for terms and conditions: some providers permit themselves to access your data to create derivative works from your data.

The data is stored in 'the cloud', often on servers outside of the United Kingdom. This raises many concerns, including privacy, security, ethical, legal and long-term preservation issues. For projects with low-risk data this may be okay, but some projects place very strict requirements on what can be done with the data especially where it is personal or sensitive data and may even stipulate that it can not leave the University.

The provisions of the Data Protection Act 1988 in respect of the collection, storage, use, disposal and transfer of personal data should always be considered.

Think carefully about the data you store in these services, investigate the encryption or the anonymisation of personal or sensitive information, and give thought to what would happen if the provider went into administration (Venkatraman, 2013), or if your account was hacked and the data was leaked.

These Dropbox-like services should not be relied on for backups as it is not easy to restore all files to a specific point in time to recover from corruption which is something you may require from a backup solution. They are useful for keeping files synchronised between machines and could help minimise data loss in the case of theft.

For synchronisation, file sharing and collaborative working without reliance on the cloud, consider peer-to-peer alternatives such as Resilio Sync (used to be called BitTorrent Sync but is now a separate, commercial offering) and the open-source Syncthing.

Office 365

Staff and postgraduates at the University also have access to Office 365 – providing the Microsoft Office Online applications and Microsoft Office software for personal equipment – but this also gives 1 TB of file storage in OneDrive for Business. This is preferable to something like Dropbox because Microsoft uses servers in the European Union but it does come with some limitations (see panel).

OneDrive for Business limitations

- Maximum 10 GB file size
- Restrictions on characters in the file name, such as files containing '`#`' or ending in '`.tmp`', or a '`forms`' folder at root level (Microsoft, 2016)

3 Version management

When dealing with large amounts of files, and changes are being made, it is important to follow a process that allows you to cope with all the different versions of the files. For example, you may want to try making a change to a CFD mesh to investigate the effect it has on the simulation. These changes may not have the desired effect so it may be necessary to 'roll-back' to an earlier version. There are a few techniques to doing this and it is important to pick one that works for you and allows you to manage your research data in a way you are comfortable with but still protects you.

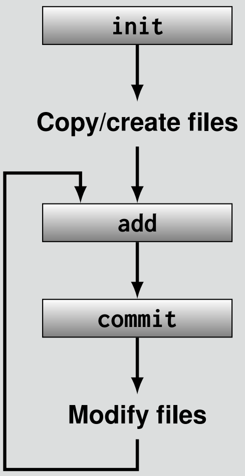
- Backup your files before editing to permit you to make a simple roll-back if necessary
- Store multiple copies of your files using some naming convention to permit easy identification of the different versions

When dealing with large amounts of files which have frequent and complex changes using specialist software that handles versioning of your files for you offers a more robust solution:

- Good examples are *Mercurial*, *Subversion* and *Git*. There are many views on which is better. Git and Mercurial use similar approaches and are probably the most lightweight and the easiest to get going with. Subversion has a steeper learning curve but it is common and the University hosts a Subversion server which may be preferable to installing it yourself.

  The same warnings given earlier about third party storage apply to repository hosting services like GitHub, Bitbucket, SourceForge and CodePlex.

Figure 20: Typical workflow for modifying a file in Git



The `init` command initialises a directory so it can track file versions

The `add` command starts tracking files

The `commit` command records a message detailing the changes along with the new versions of the added files

Mercurial is very similar to this. With Mercurial, the `add` command is only required when first adding a file to the repository and then only `commit` is required to save the file when it is modified.

4 Data security

Depending on the nature of your research and any requirements linked to your funding, the following questions may be critical:

- What restrictions have been put on your data?
- Does it need to be anonymised?
- Can it be shared?
- Does data require destroying at the end of its life?
- Are there any ethics procedures that you need to be aware of?

Encryption and data disposal are discussed below.

Encryption

If your computer is lost or stolen, data encryption will protect your data by preventing people without the passphrase from accessing your data. This is particularly important if you use a laptop and remove it from the University premises which increases the chance of the laptop being misplaced.

Encrypt your hard disk using *BitLocker* in Windows and *FileVault 2* on a Mac. For Linux, start by looking at *LUKS* for partition encryption and *EncFS* to encrypt directories.

Data disposal

When data is deleted from a disk it is not actually destroyed until it is overwritten by another file. Until then it can still be recovered. Encryption provides an extra layer of protection because, if you have encrypted your hard disk partition, then deleted data will not be recoverable unless it can first be decrypted. Data can be securely erased at the end of its lifetime, preventing future recovery – software utilities include:

- Darik's *Boot and Nuke* (DBAN) boot CD or *Parted Magic*
- Windows: *Eraser*
- Mac: *Secure Empty Trash* or *Disk Utility*
- Linux: *shred*, *srm* or *nwipe*

- With a Solid State Drive (SSD) – a disk which stores data in non-volatile memory for performance and power improvements – secure erasing in this way will seriously shorten its life and may not effectively destroy data; use the manufacturer's software to reset all blocks, or consider encryption or physical destruction.
- In some cases, physical destruction may be more reliable – contact iSolutions.

5 Data curation and preservation

EPSRC expect research data to be described online **within 12 months** of the end of data collection.

Publishing metadata about the data online is for discovery purposes and it is only the metadata record that is expected to be published immediately. The dataset itself is to be made 'freely and openly available with as few restrictions as possible in a timely and responsible manner'.

Other funding bodies have different policies but all UK Research Councils follow the RCUK's common principles on research data which encourages open practices, but recognises legal, ethical and commercial constraints. It is imperative that you understand your responsibilities based on your funder's expectations.

Data curation steps

The following steps will help others to find your data again, and ensure that your research is compliant with funders' policies:

- Create a metadata record in a data repository. This should be at the same time as you publish a paper that uses the data; otherwise it should be within 12 months of the end of data collection.

  You should describe what the data is, why, when and how it was generated, and how it can be accessed or obtained. Use EPrints if your discipline does not have its own repository.

- Obtain a Digital Object Identifier (DOI – a permanent and unique identifier) for the data record. If using EPrints, contact eprints@soton.ac.uk if you don't receive one automatically.

- If you can, upload the data with the metadata record. Use EPrints if your discipline does not have its own repository.

- Ensure the metadata for your dataset is complete and accurate.

- Include a data access statement in any published work linking to the dataset (see the following section).

Commercially confidential data is expected by many funders to still have metadata published, giving the reason why the data is restricted and the conditions that must be met in order to gain access to the dataset.

Data access statements

It is now a requirement of many funders, such as the EPSRC, to include a statement in any publications describing where the data that supports a publication can obtained. Here are some examples, which can also be viewed at `http://library.soton.ac.uk/researchdata`:

Openly available data

For openly available data, include the following in the data access statement:

- Name(s) of the data repositories
- Persistent identifiers or accession numbers for the dataset

For example:

- 'All data supporting this study are openly available from the University of Southampton repository at http://dx.doi.org/10.5258/SOTON/xxxxx.'

Real-world example:

'Data published in this paper are available from the University of Southampton repository at 10.5258/SOTON/379558.'

From: G. Squicciarini, M.G.R. Toward and D.J. Thompson (2015). 'Experimental procedures for testing the performance of rail dampers'. In: *Journal of Sound and Vibration* 359, pp. 21–39. ISSN: 0022-460X. DOI: `http://dx.doi.org/10.1016/j.jsv.2015.07.007`

Restricted access – ethical, legal, commercial

- Include justification for restriction
- Document reasons, for example:
  - the ethics approval reference number in metadata
  - collaborative agreements
  - data management plan for the project

For example:

- 'Due to ethical concerns, supporting data cannot be made openly available. Further information about the data and conditions for access are available from the University of Southampton repository: http://dx.doi.org/10.5258/SOTON/xxxxx'
- 'Bona fide researchers, subject to registration may request supporting data via University of Southampton repository http://dx.doi.org/10.5258/SOTON/xxxxx'

Real-world example:

'The study data are not freely available due to legal restrictions, and Government of India's Health Ministry Screening Committee (HMSC) assessment is required to obtain the data. The Parthenon Cohort team will provide the data on request subject to HMSC approval. For further information contact the corresponding author.'

From: Ghattu V. Krishnaveni et al. (2015). 'Linear Growth and Fat and Lean Tissue Gain during Childhood: Associations with Cardiometabolic and Cognitive Outcomes in Adolescent Indian Children'. In: *PLoS ONE* 10.11, pp. 1–14. DOI: `10.1371/journal.pone.0143231`

Secondary analysis of existing data

If your dataset manipulates/re-uses an existing dataset (derived data), the original source(s) should be credited.
For example:

- 'This study was a re-analysis of existing data that are publicly available from [organisation] at [web address]'

Real-world example:

For an example, see: Elaine L. McDonagh et al. (2015). 'Continuous Estimate of Atlantic Oceanic Freshwater Flux at 26.5 degrees N'. in: *JOURNAL OF CLIMATE* 28.22, pp. 8888–8906. ISSN: 0894-8755. DOI: `10.1175/JCLI-D-14-00519.1`

No new data created

Sometimes no datasets may be created, but you should still include a data access statement. For example:

- 'No new datasets were created during this study'

Part IV
Acknowledgements

- The categorisation of research data collection was defined in Research Information Network, 2008.
- The forms of research data and categorisation of electronic storage of research data was adapted from The University of Edinburgh, 2011.
- The following people helped with the preparation of this document:
  - Andy Collins (Human Genetics case study)
  - Thomas Mbuya and Kath Soady (Materials Fatigue Test case study)
  - Gregory Jasion (CFD case study)
  - Simon Coles (Chemistry case study)
  - Graeme Earl (Archaeology case study)
- The original work was supported by the University of Southampton, Robert's funding, the JISC DataPool project, Microsoft, EPSRC, BBSRC, JISC, AHRC and MRC. We acknowledge ongoing support from Microsoft and EPSRC.

References

Broad Institute (2016). *Integrative Genomics Viewer*.
URL: http://www.broadinstitute.org/software/igv/ (visited on 04/10/2016).

Bullas, John C (2005). *That's where my office was last time I looked!* CC BY-NC-ND 2.0.
URL: https://www.flickr.com/photos/johnbullas/58458915/in/album-1247706/ (visited on 04/10/2016).

Digital Curation Centre (2016). *DCC Curation Lifecycle Model*.
URL: http://www.dcc.ac.uk/resources/curation-lifecycle-model (visited on 04/10/2016).

Humphrey, Charles (2006). *e-Science and the Life Cycle of Research*.
URL: http://datalib.library.ualberta.ca/~humphrey/lifecycle-science060308.doc (visited on 20/02/2012).

Microsoft (2016). *Restrictions and limitations when you sync OneDrive for Business libraries through OneDrive for Business*. Knowledge Base.
URL: https://support.microsoft.com/kb/3125202 (visited on 04/10/2016).

Research Information Network (2008). *Stewardship of digital research data: a framework of principles and guidelines*.
URL: http://www.rin.ac.uk/our-work/data-management-and-curation/stewardship-digital-research-data-principles-and-guidelines (visited on 02/02/2015).

The University of Edinburgh (2011). *Defining research data*.
URL: http://www.ed.ac.uk/schools-departments/information-services/services/research-support/data-library/research-data-mgmt/data-mgmt/research-data-definition (visited on 20/02/2012).

University of Southampton (2016). *Portus Project*.
URL: http://www.portusproject.org/ (visited on 04/10/2016).

University of York (2016). *Archaeology Data Service*.
URL: http://archaeologydataservice.ac.uk/ (visited on 04/10/2016).

Venkatraman, Archana (2013). *2e2 datacentre administrators hold customers' data to £1m ransom*. Computer Weekly.
URL: http://www.computerweekly.com/news/2240177744/2e2-datacentre-administrators-hold-customers-data-to-1m-ransom (visited on 04/10/2016).