Chapter 5

Data journalism

This chapter will cover:

- Data journalism in action
- Finding data
- Gathering data yourself
- Interrogating the data
- Closer look: Cleaning up data
- Closer look: Quickly generating averages and percentages using spreadsheets
- Closer look: Tips from a pro: David McCandless
- Visualising data
- Mashing data
- Closer look: How to publish a spreadsheet online
- Closer look: Using Yahoo! Pipes to create a sports news widget
- Closer look: Taking it further – resources on programming
- Closer look: Anatomy of a feed

Introduction

> '[*Telegraph* journalist Holly] Watt had set up a spreadsheet which listed MPs' names, the addresses of properties on which they had claimed, whether each was rented or mortgaged, and notes of other points about their claims. By the middle of the second week . . . she had a potential breakthrough. As she typed in the address of the second home of Ian Cawsey, an obscure Labour MP, a box popped up on her screen with the rest of the address. This 'auto-complete' function meant she had already typed in the address for another MP.
>
> '"Bingo!" Watt shouted across the office.
>
> 'Scrolling back up the spreadsheet, Watt found that Cawsey's London address was shared with [another MP] . . . In other words, the taxpayer was being billed twice for the same property.'
>
> *No Expenses Spared*,
> Robert Winnett & Gordon Rayner, 2009, p. 220

The biggest political story of the 21st century was, at its heart, about data. A disk containing the expenses claims of hundreds of MPs landed on a desk in Buckingham Palace Road – and *Telegraph* journalists had less than a week to decide if, somewhere inside the millions of documents it contained, there was a story worth paying for. The decision that there was saw the newspaper dominate the news agenda for the following six weeks, thanks to a combination of traditional news-gathering skills with a newer skillset: the ability to find the story behind the data.

Journalists have always had to deal with certain types of information. From eyewitness accounts and official documents, to research papers, reports and press releases – part of a journalist's talent has been their ability to cut through the fat to the meat of the story.

The potential of databases for news gathering was identified as early as the 1960s, when US reporter Philip Meyer used them to investigate race riots. In 1973 Meyer published a seminal work in the field, *Precision Journalism*. The practices established in that book, and practised by others, became known as computer-assisted reporting, and the field gained an institution of its own in 1989 with the founding of the National Institute for Computer-Assisted Reporting (NICAR).

However, as the profession moved into the 21st century the focus broadened – from databases (private, inaccessible, few) to data (public, accessible, many). Joe Hellerstein, a computer scientist at the University of California in Berkeley, calls it 'the industrial revolution of data'. Others call the phenomenon 'big data' (Cukier, 2010).

'Data' has always been a vague term. To some, it refers to statistics and facts; to others, it is something specific: structured information in a particular format. For the purposes of this book, it is perhaps best defined as 'information that can be analysed with computers'. This can range from numbers to reams of text (what are the most common terms used?), images and video (which colours and patterns recur most?), and even behavioural data (when are people most active? Where do they cluster online?). If information has been digitised, then there are patterns to be found there.

But, however you define data, it is clear that there is an increasing amount of it around. Spreadsheets and databases have become a part of everyday life in business, government, and even entertainment and sport – and increasingly widespread publication of these online is proving to be a goldmine for those who know how to use them, whether analysing football results or investigating the awarding of government contracts.

At the same time, a number of leading figures on the World Wide Web, including its inventor Tim Berners-Lee, have been leading a movement to make it easy for computers – and people – to make links between various sets of data.

In particular, this has produced increasing pressure on governments to release data about their activities to the public, and to do so in a way that made it as easy as possible to interrogate (so, for example, you might be able to quickly spot that Politician X and the son of Politician X worked for the same company).

Coinciding with – if not preceding – this 'open data' movement has been the introduction of a series of Freedom of Information laws around the world that have given citizens the right to access data generated by public bodies, from crime information to details on how public money is spent; from inspection reports to information held about yourself.

As governments have released data, bloggers, scientists and web developers have been among the first to start doing interesting things with it – distributing it, analysing it, and 'mashing' it up with other sources to display it on a map, for example, or in comparison with other data sets. As these pioneers streaked ahead, journalists and news organisations were being left behind, but by early 2009 a vanguard of publishers had developed with coherent strategies around data-driven journalism.

Data journalism may be a new term, but the idea of ploughing through information to find the story is as old as journalism itself – the difference is that the internet and computing power are giving us new tools to help at every stage of the journalistic process.

Whether it's finding data in the first place, or interrogating it in order to find a story; whether you are visualising data in meaningful ways to tell a story, combining it with other information to create new insights, or simply releasing it in a way that makes it as easy as possible for users to do their own digging, data journalism is a key feature of the future of journalism on all platforms, but particularly online.

This chapter explores all these stages of data journalism, looking at some of its most high-profile examples, and providing practical advice on how to start developing your own data journalism skills.

It is the most challenging chapter in this book, but the lessons it contains are perhaps the most important for journalism's future, because data journalism is one of the key ways in which journalists and citizens can expose ignorance and challenge prejudice, make complex facts accessible, and hold power to account. And that, after all, is what journalism should be about.

Data journalism in action

> 'More than 12,000 flight plans were now stored in my computer. I was trying to narrow things down – find the pattern that lurked beneath all this data and the identity of the CIA's planes that might be involved in rendition. When I started my investigation, I had almost no information but now I was almost swamped. I turned to a software programme called Analyst's Notebook, a tool used normally by the police or intelligence organisations to solve complex financial crimes or even murders. Its job was to find connections within exactly such a mass of data.'
>
> *Ghost Plane: The Inside Story of the CIA's Secret Rendition Programme*, Stephen Grey, 2006, p. 107

Data journalism takes many forms. Sometimes the data itself tells the story so clearly that the journalist's job is merely to present it in such a way that its meaning is as clear as possible, or so that users can add to it. Sometimes the data is the starting point for an investigation that takes the journalist to meet others who can shed light on the questions it raises. And sometimes the data does not exist anywhere, and the journalist must gather it by meeting sources and visiting libraries, until they have enough to put together the data they need.

One of the pioneers of modern data journalism is Adrian Holovaty. Holovaty launched ChicagoCrime.org in 2005 – a 'mashup' which placed public crime data on a map of the city and instantly provided citizens with an important source of information on local crime. The idea was widely imitated by other newspapers, and ChicagoCrime eventually received funding to relaunch as EveryBlock (everyblock.com), which extended the concept to show planning applications, alcohol licences, restaurant inspections and street closures – among other pieces of data. It has also expanded to cover more than a dozen US cities – and with the technology behind the site released as 'open source', this means that anyone else can use it for other cities, anywhere in the world. Meanwhile, other news operations – such as MSN Local in the UK (local.uk.msn.com) – have adopted similar data-driven approaches.

Holovaty sees journalism as involving three core tasks – gathering, distilling and presenting information. Doing journalism through computer programming – just one form of data journalism – is, he says, just a different way of doing those tasks:

> 'Each weekday, my computer program goes to the Chicago Police Department's website and gathers all crimes reported in Chicago. Similarly, the U.S. Congress votes database I helped put together at washingtonpost.com works the same way: several times a day, an automated program checks several government websites for roll-call votes. If it finds any, it gathers the data and saves it into a database.
>
> '[And] just as an editor can apply editorial judgment to decide which facts in a news story are most important . . . on chicagocrime.org I decided it would be useful if site users could browse by crime type, ZIP code and city ward. On the votes database site, we decided it would be useful to browse a list of all the votes that happen late at night and a list of members of Congress who've missed the most votes. Once we made that decision of which information to display, it was just a matter of writing the programming code that automated it.
>
> 'Presentation is also automated. This is particularly complex, because in creating websites, it's necessary to account for all possible permutations of data. For example, on chicagocrime.org I had to account for missing data: how should the site display crimes whose data has changed? What should happen in the case where a crime's longitude/latitude coordinates aren't available? What should happen when a crime's time is listed as "Not available"?'
>
> 'The programmer as journalist: a Q&A with Adrian Holovaty', *Online Journalism Review*, Robert Niles, 2006

Although Holovaty's project involved programming, it is also possible to create similar, more basic mashups with just a few clicks, thanks to free web services like Yahoo! Pipes (see below).

Sometimes data journalism involves accessing or creating databases of information. In *Ghost Plane*, the story of Stephen Grey's investigation into a CIA practice of flying terror suspects to countries where they would be tortured, the *Sunday Times* investigative journalist talks about his global journey to visit sources as he gathered the evidence he needed to put together a picture of 'extraordinary rendition'. Data provided a key element of his investigation – specifically, databases of flight plans (which allowed him to verify the accounts of terror suspects who had been transported) and of people (which allowed others at the *New York Times* to establish the creation of fake identities that masked the existence of an airline run by the CIA).

At the BBC, Dominic Casciani drew on immigration data to produce a website looking at the subject (Casciani, 2005). Like much data journalism, the site allowed users to interrogate the issue in their own way, while also providing an editorial overview. In a similar vein *The Telegraph* added to its MPs' expenses database to create a powerful tool that readers could use to find out about all candidates in the run-up to the 2010 election (Bradshaw, 2010), while Channel 4 created 'Who knows who' to show 'the connections between politicians, celebrities and business leaders, and where power really lies in the UK.' Similar ideas have been developed by Silobreaker's Network (silobreaker.com/FlashNetwork.aspx?q=) and They Rule in the USA (www.theyrule.net).

Again, while these projects involved building in-house databases, there are free tools – such as DabbleDB and even Google Docs – that allow anyone to do the same.

This is becoming increasingly important as news organisations and journalists look to users of their website to help dig deeper into a story. When the official MPs' expenses documents were released by parliament, for example, *The Guardian* published them all online and invited readers to help spot possible stories and to build a more effective database, which both readers and journalists could then scrutinise. The same paper had earlier published MPs' travel expenses which Tony Hirst, publisher of the blog OUseful, visualised on a map in a way that made it easy to tell at a glance which MPs were claiming more for travel than other MPs who lived nearby (Arthur, 2009a).

Maps are particularly useful devices for the data journalist, especially when they are dealing with information that includes latitude and longitude, a postcode or placename. During one US election, for example, a number of map-related 'mashups' appeared showing Twitter tweets, YouTube videos or other election-related material displayed on a Google map. During the Beijing Olympics BBC Sport used similar technology to display tweets, blog posts and photos on a map of the Olympic village (Hamman, 2008).

Data is particularly useful in sport, where statistics and location are often both important and real-time information is in high demand. BBC Sport experimented with one 'widget' that combed a selected group of feeds for the latest news during the last days of the football transfer window. This chapter will explain how you can create a similar mashup using Yahoo! Pipes.

Before that, the chapter will seek to provide an overview of four basic stages that a data journalism project may go through: finding data; interrogating it; visualising it; and mashing it (Figure 5.1). The field of data journalism is so diverse that any one project might take in only one of the stages – finding information to share easily with users, for example. But the more stages you experience, the wider you read on the methods involved, and the more you experiment, the better your journalism will be.

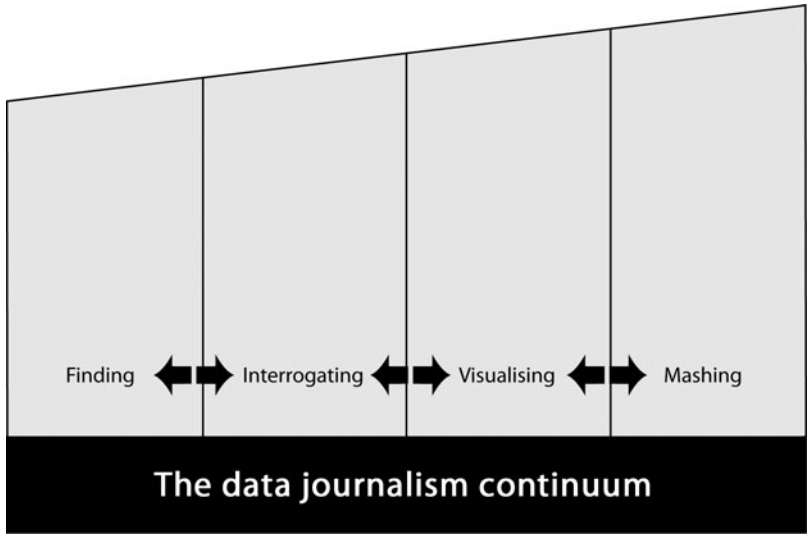


Figure 5.1 The data journalism continuum

Finding data

The first stage in data journalism is sourcing the data itself. Often you will be seeking out data based on a particular question or hypothesis (for a good guide to forming a journalistic hypothesis, see Mark Hunter's free ebook *Story-Based Inquiry* (2010)). On other occasions, it may be that the release or discovery of data itself kicks off your investigation.

There is a range of sources available to the data journalist, both online and offline, public and hidden. Typical sources include:

- national and local government;
- bodies that monitor organisations (such as regulators or consumer bodies);
- scientific and academic institutions;
- health organisations;
- charities and pressure groups;
- business;
- and the media itself.

One of the best places to find UK government data online, for example, is Data.gov.uk, an initiative influenced by its US predecessor Data.gov. Data.gov.uk – launched in January 2010 with the backing of the inventor of the World Wide Web, Sir Tim Berners-Lee – effectively acts as a search engine and index for thousands of sets of data held by a range of government departments, from statistics on the re-offending of juveniles to the Agricultural Price Index. The site also hosts forums for users to discuss their use of the data, examples of applications using data, further information on how to use the data, and technical resources. For archive material NDAD (the National Archives' National Digital Archive of Datasets) is worth a visit at www.ndad.nationalarchives.gov.uk.

At a regional and local level, Oneplace (oneplace.audit-commission.gov.uk) was a good source of information (the assessments upon which the data was based ended in June 2010), while individual local authorities are also releasing information that can be used as part of data journalism projects. The quality and quantity of this information varies enormously by council, but there is continuing pressure for improvement in this area.

There are also volunteer projects, such as OpenlyLocal and Mash The State, that make local government data available in as accessible a format as possible, while the organisation MySociety operates a group of websites providing easy access to information ranging from individual politicians' voting record (TheyWorkForYou) to local problems (FixMyStreet) and information about a particular area's transport links and beauty (Mapumental). MySociety also runs a petitions website for Downing Street, and websites that allow people to pledge to do something if other people sign up too (PledgeBank), to find groups near you (GroupsNearYou), contact your MP (WriteToThem) or be contacted by them (HearFromYourMP).

In the private sector, organisations regularly release data online, from tables and research reports published on company websites to the annual reports that are filed with bodies such as Companies House. Also worth looking at is the web project Open Corporates (opencorporates.com), which seeks to make company information more easily accessible.

The Charity Commission is an excellent source of information on registered charities, which must file accounts and annual reports with the organisation. The commission also conducts occasional research into the sector.

NHS foundation trusts likewise must file reports to their regulator, Monitor. You will find similar regulators in other areas such as the Financial Services Authority, Ofcom, Ofwat, Ofqual, the General Medical Council, the General Social Care Council and the Pensions Regulator, to name just a few.

For academic and scientific research there are hundreds of specialist journals. Most have online search facilities which will provide access to summaries. To get access to the full paper you will probably need to use the library of a university, which involves a subscription. For access to a journal on midwifery, for example, your best bet is to make a quick call to the nearest university that teaches courses in that field. Although university libraries increasingly limit access to students, you can request a special pass. For access to the data on which research is based it is likely you will need to contact the author. Also of potential use in this area is the UK Data Archive (data-archive.ac.uk).

Media organisations such as *The Guardian* and *The New York Times* publish 'datablogs' that regularly release sets of data produced or acquired by investigations, ranging from scientific information about global warming to lists of Oscar winners. These can be a rich source of material for the data journalist, and a great starting point for the beginner as they are often 'cleaner' than data from elsewhere. Also worth investigating are websites such as WikiLeaks and ScraperWiki, which provide a repository for sets of data.

The Guardian and *The New York Times* websites are also among an increasing number of web platforms generally which are making their own data available via APIs (application programming interfaces). Social networking sites (such as Flickr and Twitter) also provide APIs.

Accessing this data generally requires a level of technical ability, but can be particularly useful in measuring activity across social networks (for example, sharing and publishing). Even if you don't have that technical ability, understanding the possibilities can be extremely useful when working with web developers on a data journalism project (see the part of this chapter on mashups for more information on APIs).

Using search engines to find data

If you are using a search engine to find the data you are looking for, you should familiarise yourself with the advanced search facility, where you can often specify the format of the file you are looking for. Searching specifically for spreadsheets (files ending in .xls), for example, is likely to get you to data more quickly. Similarly, official reports can often be found more effectively by searching for PDF format, while PowerPoint presentations (.ppt) will sometimes contain useful tables of data. You can also include 'XML' or 'RDF' in your search terms if you think your data may be in those or other formats.

Advanced search also allows you to specify the type of website you are searching for – those ending in .gov.uk (government), .org and .org.uk (charities), .ac.uk (educational establishments), .nhs, .police.uk and .mod (Ministry of Defence) are just some that will be particularly relevant (you can also specify an individual site – for instance, that of a local council). A basic familiarity with these search techniques – for example, limiting your search to spreadsheets on .gov.uk websites – can improve your results.

To use advanced search, look for an 'advanced search' link. Or you can use particular syntax – for example, adding `site:gov.uk` to a Google search will limit results to sites ending in .gov.uk; likewise `site:bolton.gov.uk` would limit it to the Bolton Council website. Adding `filetype:pdf` will limit results to PDFs, and so on. For an extensive guide to Google search, see: www.googleguide.com/advanced_operators.html.

You should, however, remember that Google only searches a small part of the internet and there may be other search engines that give better results for different searches. Wolfram Alpha, for example, is a 'computational knowledge engine' that searches a range of data sources from around the world, allowing you to type intelligent queries such as 'UK GDP 2000–2009' and see where the data is pulled from by clicking on 'Source information' below the results. There are also various specialist 'vertical' search engines such as Microsoft Academic Search (http://academic.research.microsoft.com/).

Live data

Another type of data to think about is live data that is not stored anywhere yet but, rather, will be produced at a particular time. A good example of this would be how newspapers are increasingly using Twitter commentary to provide context to a particular debate. Part of *The Guardian*'s coverage of Tony Blair's appearance at the Chilcot Inquiry into the Iraq War, for example, used the data of thousands of Twitter updates ('tweets') to provide a 'sentiment analysis' timeline of how people reacted to particular parts of his evidence as it went on (*The Guardian*, 2010a, 2010b). Similar timelines have been produced for political debates and speeches to measure public reaction.

Preparation is key to live data projects – where will you get the data from, and how will you filter it? How can you visualise it most clearly? And how do you prevent it being 'gamed' (users intentionally skewing the results for fun or commercial or political reasons)?

Legal considerations

Whatever data you are acquiring, you will need to consider whether you have permission to republish that data (analysing the data privately in the first place is a separate issue and unlikely to raise legal issues).

Data may be covered by copyright, or may raise issues of data protection or privacy. Even apparently anonymous information can sometimes be traced back to individual users (Barbaro & Zeller, 2006), and while government information is paid for by public money, for example, it is, strictly speaking, often covered by Crown Copyright, while organisations like Ordnance Survey

and Royal Mail have been notoriously protective of geographical information and postcodes (see Heather Brooke's book *The Silent State*, 2010, for more on the tactics used by organisations to prevent access to 'public' data).

In addition, if you are using free online tools to visualise or share your data (see below), you may want to check the terms and conditions of those services in case they conflict with the terms under which you obtained the data or intend to use it.

Books and Freedom of Information (FOI)

Of course, there is also a rich range of data available in books that data journalists should familiarise themselves with – from books of facts and statistics to almanacs, from the Civil Service Year Book (also online) to volumes like Who's Who (online at ukwhoswho.com – your library may have a subscription).

Particularly useful is the data held by public bodies which can be accessed through a well-worded Freedom of Information (FOI) request. Heather Brooke's book *Your Right To Know* (2007) is a key reference work in this area, and the online tool WhatDoTheyKnow is particularly useful in allowing you to submit FOI requests easily, as well as allowing you to find similar FOI requests and the responses to them. In addition, you should look to see if an organisation publishes a disclosure log of the FOI requests it has received (tip: use Google's Advanced Search to look for 'disclosure log' within a particular website or domain such as .gov.uk).

When requesting data through an FOI request, it is always useful to specify the format that you wish the information to be supplied in – typically a spreadsheet in electronic format. A PDF or Word document, for example, will mean extra work at the next stage: interrogation.

Gathering data yourself

Of course, the data you need for a particular story may not exist at all in either books or online – or the data you do find is out of date, flawed, scattered across several sources, or needs checking. In that case you'll need to gather data yourself.

There are two basic approaches you can take in gathering data: compiling it from other sources such as newspaper or other reports (secondary research), or collecting it yourself through methods such as observation or surveys (primary research).

Whichever method you use, you should understand potential weaknesses in the methodology and seek to address these. The selection, size and generalisability of the sample; the selection and phrasing of questions; the environment; the use of a control group; your own presence; and myriad other factors must be taken into account to ensure that your methods stand up to scrutiny.

Needless to say there are countless books covering research methodology and you should read these if you are to do any research of your own. Even if you don't, they will give you an understanding that will prove very useful in looking at the methods used to gather other data that you might use. For an accessible read on the subject see Ben Goldacre's *Bad Science* book (2008) and blog (www.badscience.net), or one of the increasing number of 'sceptics' blogs such as Quackometer (www.quackometer.net).

Interrogating the data

> 'One of the most important (and least technical) skills in understanding data is asking good questions. An appropriate question shares an interest you have in the data, tries to convey it to others, and is curiosity-oriented rather than maths-oriented. Visualising data is just like any

> other type of communication: success is defined by your audience's ability to pick up on, and be excited about, your insight.'
>
> Fry, 2008, p. 4

Once you have the data, you need to see if there is a story buried within it. The great advantage of computer processing is that it makes it easier to sort, filter, compare and search information in different ways to get to the heart of what – if anything – it reveals.

The first stage in this process, then, is making sure the data is in the right format to be interrogated. Quite often this will be a spreadsheet or CSV (comma-separated values) file. If your information is in a PDF you may not be able to do a great deal with it other than copy or re-type the values into a new spreadsheet (making sure to check you have not made any errors). In some cases you can use optical character recognition (OCR) and 'export to spreadsheet' to extract the data, but it's better if you don't have to face that problem.

A Microsoft Word or PowerPoint document is likely to require similar work. Note: if you're pasting into Excel it has a useful tool for cleaning up data that is in rows but not in a recognised columnar format: paste into a column and select Data > Text to Columns.

If the information is already online you can sometimes 'scrape' it – that is, automatically copy the relevant information into a separate document. How easy this is to do depends on how structured the information is. A table in a Wikipedia entry, for example, can be 'scraped' into a Google spreadsheet relatively easily (Tony Hirst gives instructions on how to do this at: http://blog.ouseful.info/2008/10/14/data-scraping-wikipedia-with-google-spreadsheets/), and an online CSV file and certain other structured data can be scraped with Yahoo! Pipes

Closer look Cleaning up data

Whether you have been given data, had to scrape it, or copied it manually, you will probably need to clean it up. All sorts of things can 'dirty' your data, from misspellings and variations in spelling, to odd punctuation, mixtures of numbers and letters, unnecessary columns or rows, and more. Computers, for example, will see 'New Town', 'Newtown' and 'newtown' as three separate towns when they may be one.

This can cause problems later on when analysing your data – for example, calculations not working or results not being accurate.

Some tips for cleaning your data include:

- Use a spellchecker to check for misspellings. You will probably have to add some words to the computer's dictionary.
- Use 'find and replace' (normally in the Edit menu) to remove double spaces and other common punctuation errors.
- Remove duplicate entries – if you are using Excel there are a few ways to do this under the Data tab – search for duplicates in Help.
- Watch out for extra spaces at the beginning and end of text; they are often easy to miss and may prevent matching in some programs.

The free tool, Google Refine, is also worth exploring.

For more tips on Excel specifically, see this guide: http://office.microsoft.com/en-us/excel/HA102218401033.aspx

(see http://www.daybarr.com/blog/yahoo-pipes-tutorial-an-example-using-the-fetch-page-module-to-make-a-web-scraper and below for more on using Yahoo! Pipes). You can also use the Firefox extension OutWit Hub which presents you with a step-by-step guide to extracting data from a webpage of your choice. A lot of scraping, however, will involve programming – the online tool ScraperWiki provides one environment to help you do this. For more information on scraping and scraping tools, see the links at: www.delicious.com/paulb/scraping.

Sometimes data is incomplete: for example, it may be lacking a person's first name or date of birth, which makes it hard to connect with other sources. For this reason and all those above, it is a good idea to develop your own databases, and publish where legally possible.

Spotting the story

Once your data is cleaned you can start to look for anything newsworthy in it. There are some obvious places to start: if you are dealing with numbers, for example, you can work out what the 'mean' is (the average age of chief inspectors, for example). Similarly, you might look for the 'mode' – the term or value which appears most often (e.g. the most common reason given for arresting terrorist suspects) or the 'median' – the middle value in the range covered. Typically, medians are best used for financial statistics such as wages, means are best used for physical characteristics, and modes are best used to illustrate the 'best seller' or 'most used'.

All of these come under the vague term 'average', and are subject to abuse (see *How to Lie With Statistics*, Huff, 1954, for more on this and other statistical tricks). Amber Iler, commenting on a draft of this chapter on the Online Journalism Blog, puts it this way:

> 'Means can be abused by averaging over time: for example, an average increase of 20 cement trucks/day on a road may not seem like a lot until you look at the median increase and see that they all use the road between 7–9 AM. You get a very different picture when you think about sharing the road with 20 trucks spread out over the course of a day than you do imagining the congestion of 20 trucks on your road during rush hour.'

Kaiser Fung, a statistician whose blog Junk Charts is essential reading in the field, notes the dangers in lazily reaching for the average when you want to make an editorial point:

> 'Averaging stamps out diversity, reducing anything to its simplest terms. In so doing, we run the risk of oversimplifying, of forgetting the variations around the average. Hitching one's attention to these variations rather than the average is a sure sign of maturity in statistical thinking. One can, in fact, *define* statistics as the study of the nature of variability. How much do things change? How large are these variations? What causes them?'
>
> Fung, 2010, p. 4

So, while averages can be interesting discoveries, they should most often be used as a starting point for more illuminating investigation.

If you are looking at data over time, you can look to see what has increased over that period, or decreased – or disappeared entirely. You will need to make sure you are expressing percentages correctly. If the amount spent by an organisation on recruitment, for example, rises from £200,000 to £250,000 this represents a 25 per cent increase; the reverse, however (£250k to £200k), would be a decrease of 20 per cent.

Calculations like these can be made easily with spreadsheet software such as Excel and the spreadsheet in Google Docs (see *Closer look: Quickly generating averages and percentages using spreadsheets* on page 58). Many journalists in the field of computer-assisted reporting (CAR) will also use database software such as Access and languages such as SQL which provide very

powerful ways to interrogate thousands or millions of lines of data. This is an area worth exploring if you become serious about this aspect of data journalism.

You will also need to gather further data to provide context to your figures. If, for example, more council staff are receiving bonuses, is that simply because more staff have been employed? How much is spent on wages, and how do your figures compare? If you are comparing one city with another, understand how their populations differ – not just in aggregate, but in relevant details such as age, ethnicity, life expectancy, etc. You will need to know where to access basic statistics like these – the National Statistics website (www.statistics.gov.uk) is often a good place to start.

As a journalist you should double-check your findings wherever possible with statisticians in the field you're covering. The Royal Statistical Society maintains a list of media contacts you can contact across sectors from finance and the environment to DNA, forensic evidence and health.

Sometimes a change in the way data is gathered or categorised can produce a dramatic change in the data itself. In one example, designer Adrian Short obtained information (via an FOI request) on parking tickets from Transport for London that showed the numbers of tickets issued against a particular offence plummeted from around 8,000 to 8 in the space of one month (Arthur, 2009b). Had people suddenly stopped committing that parking offence, or was there another explanation? A quick phone call to Transport for London revealed that traffic wardens

Closer look Quickly generating averages and percentages using spreadsheets

Spreadsheet packages can save you a lot of time as a journalist. Excel and the free spreadsheet software in Google Docs can quickly generate an average or percentage from sheets of numbers using formulae. Here are a few useful formulae that you can adapt for your own data (some spreadsheet packages may use different formulae):

=average(a2:a300)

This will generate a mean from all the numbers between cells A2 and A300 (A is the first column; the number 2 refers to the row – change these to fit where your own data is in the spreadsheet).

=median(a2:a300)

This will give you the median value of all the numbers between cells A2 and A300.

=max(a2:a300)

This will give you the maximum number within that range.

=countif(a2:a300,140)

This will count how many times the figure '140' is mentioned in the numbers between cells A2 and A300.

=(b2-a2)/a2

This will calculate a percentage increase between two figures, as long as B2 is the new figure and A2 is the old figure.

To find formulae for other calculations just do a quick search on what you want to do and the word 'spreadsheet formula'.

Closer look Tips from a pro: David McCandless

David McCandless is a writer and designer, and author of the book and website *Information is Beautiful* (2010). His work has appeared in *The Guardian*, *The Independent* and *Wired* magazine. These are his five tips for visualising data:

1. Double-source data wherever possible – even the UN and WorldBank can make mistakes.
2. Take information out – there is a long tradition among statistical journalists of showing *everything*. All data points. The whole range. Every column and row. But stories are about clear threads with extraneous information fuzzed out. And journalism is about telling stories. You can only truly do that when you mask out the irrelevant or the minor data. The same applies to design, which is about reducing something to its functional essence.
3. Avoid standard abstract units – tons of carbon, billions of dollars – these kinds of units are overused and impossible to imagine or relate to. Try to rework or process units down to 'everyday' measures. Try to give meaningful context for huge figures whenever possible.
4. Self-sufficiency – all graphs, charts and infographics should be self-sufficient. That is, you shouldn't require any other information to understand them. They're like interfaces. Each should have a clear title, legend, source, labels, etc. And credit yourself. I've seen too many great visuals with no credit or name at the bottom.
5. Show your workings – transparency seems like a new front for journalists. Google Docs makes it incredibly easy to share your data and thought processes with readers – who can then participate.

were issued with new handsets around the same time. *Guardian* journalist Charles Arthur hypothesised:

> 'Could it be that s46 [another offence which had a steep rise at the same time] is the default on the screen to issue a new ticket, and that wardens don't bother to change it? Whatever it is, there's a serious problem for TfL if those aren't all s46 offences which have been ticketed since August 2006. Because if the ticket isn't written out to the correct offence, then the fine isn't payable. Theoretically, TfL might have to pay back millions in traffic fines for people who have been ticketed for s46 offences when they were actually committing s25 or s30 offences.'

That particular story came about at least in part because the information was easy to visualise.

Visualising data

> 'At their best, graphics are instruments for reasoning about quantitative information. Often the most effective way to describe, explore, and summarise a set of numbers – even a very large set – is to look at pictures of those numbers.'
>
> Edward Tufte, *The Visual Display of Quantitative Information, 2001*, p. 9

Visualisation is the process of giving a visual form to information which is often otherwise dry or impenetrable. Traditional examples of visualisation include turning a table into a bar chart, or a

series of percentage values into a pie chart – but the increasing power of both computer analysis and graphic design software has seen the craft of visualisation develop with increasing sophistication, adding animation and personalisation, among other things. More recently, the spread of social media and the widespread distribution of 'Big Infographics' (Yau, 2010) have also added to their popularity.

In larger organisations the data journalist may work with a graphic artist to produce an infographic that visualises their story, but in smaller teams, in the initial stages of a story, or when speed is of the essence they are likely to need to use visualisation tools to give form to their data.

Broadly speaking, there are two typical reasons for visualising data: to find a story or to tell one. Quite often, it is both.

In the parking tickets story above it was the process of visualisation that tipped off Adrian Short and *Guardian* journalist Charles Arthur to the story – and led to further enquiries.

In most cases, however, the story will not be as immediately visible. Sometimes the data will need to be visualised in different ways before a story becomes clear. An understanding of the strengths of different types of visualisation can be particularly useful here.

Types of visualisation

Visualisation can take on a range of forms. The most familiar are those we know from maths and statistics: *pie charts*, for example, allow you to show how one thing is divided – for example, how a budget is spent, or how a population is distributed. Pie charts are thought to be particularly useful when the proportions represented are large (for example, above 25 per cent), but less useful when lower percentages are involved, due to issues with perception and the ability to compare different elements. Slices in pie charts should be ordered clockwise, from largest to smallest, and there should not be any more than five slices.

More useful for lower percentages are *bar charts* or *histograms*. Although these look the same there are subtle differences between them: the bars in bar charts represent categories (such as different cities), whereas bars in histograms represent different values on a continuum (for instance, ages, weights or amounts). The advantage of both types of chart over pie charts is that users can more easily see the difference between one quantity and another. Histograms also allow you to show change over time. You should avoid using 3D or shadow effects in bar charts as these do not add to the information or clarity.

Pictograms are like bar charts but use an icon to represent quantity – so a population of 50,000 might be represented by five 'person' icons. It is not advisable to use pictograms if quantities are close together as the user will find it harder to discern the differences.

Also useful for showing change over time are *line graphs*. Lines are 'suited for showing trend, acceleration or deceleration, and volatility, including sudden peaks or troughs' (Wong, 2010, p. 51). In addition, a series of lines overlaid upon each other can also quickly show if any variables change at different points or at simultaneous points, suggesting either relationships or shared causes (but by no means proving it – these should be taken as starting points for further investigation. You should also avoid plotting more than four lines in one chart for purposes of clarity.). Line graphs should not be used to show unrelated events, such as the test scores of a group of people.

Scattergrams are similar to line graphs, showing the distribution of individual elements against two axes, but can be particularly useful in showing up 'outliers'. Outliers are pieces of data which differ noticeably from the rest. These may be of particular interest journalistically when they show, for example, an MP claiming substantially more (or less) expenses than their peers.

A number of charts can be visualised together in what are sometimes called ***small multiples***, enabling the journalist or users to display a number of pie charts, line graphs or other charts alongside each other – allowing comparison, for example, between different populations.

Two increasingly popular forms of visualisation online are treemaps and bubble charts. Unlike other charts which allow you to visualise two aspects of the data (i.e. their place on each axis) ***bubble charts*** allow you to visualise three aspects of the data, the third being represented by the size of the bubble itself. A particularly good example of bubble charts in action can be seen in Hans Rosling's TED talk on debunking third-world myths (www.youtube.com/watch?v=RUwS1uAdUcI) – a presentation which also demonstrates the potential of other forms of visualisation, and animation, in presenting complex information in an easy-to-understand way. You can recreate Rosling's talk and play with visualising similar data using Gapminder (www.gapminder.org).

Treemaps visualise hierarchical data in a way that might best be described as rectangular pie charts-within-pie charts. This is particularly useful for representing different parts of a whole and their relationship to each other, for instance, different budgets within a government.

Perhaps the best-known example of a treemap is Newsmap (http://newsmap.jp/), created in 2004 by Marcos Weskamp (Plate 2). This visualises the amount of coverage given to stories by news organisations based on a feed from Google News. Weskamp explains it as follows:

> 'Google News automatically groups news stories with similar content and places them based on algorithmic results into clusters. In Newsmap, the size of each cell is determined by the amount of related articles that exist inside each news cluster that the Google News Aggregator presents. In that way users can quickly identify which news stories have been given the most coverage, viewing the map by region, topic or time. Through that process it still accentuates the importance of a given article.'
>
> Weskamp, 2005

More broadly, *maps* allow you to visualise information geographically. This can be done with points (such as the locations of crimes in an area) or lines (the routes taken) or with colouring – for example, illustrating how often crimes are committed in a particular area by giving it a colour on a spectrum from yellow to red. Maps coloured this way are called *chloropleth* maps. Care should be taken in choosing how to split data between different colour categories in a chloropleth map so as not to mislead readers.

These are examples of the most common forms of visualisation, but there are dozens more to explore.

Considerations in visualisation

When visualising data it is important to ensure that any comparisons are meaningful, or like-for-like. In one visualisation of how many sales a musician needs to make to earn the minimum wage, for example, a comparison is made between sites selling albums, sites selling individual tracks, and those providing music streams. Clearly this is misleading – and was criticised for being so (Yang, 2010).

Visualisation should also be used only when it adds something to a story. Staci Baird (2010) compares its role with that of headlines and photos in attracting readers, while warning that 'overly complex graphics will quickly convince readers to ignore your article.'

Baird advises being concise in your information design to convey one idea really well, and a good infographic, she suggests, shouldn't need a legend to be understandable. She recommends avoiding over-used chart types such as pie charts or bar charts if possible – or mixing different types together for variety – while including visuals, illustrations and photos to make visualisations more attractive. On a more practical level, you should design for the final intended viewing size (don't require users to click away from an article to see a

graphic), and always include copyright, data sources, credits and other information on the graphic itself.

The Times' Julian Burgess says the key thing with visualisation is simply 'Lots of practice.'

> 'Use stuff like Excel, so you get a feel for how graphs and visualisations work, what quantities of data you can deal with. If you're dealing with even slightly large amounts of data, then looking at using a database will be well worth it. They are much quicker and hard to screw up your data compared to spreadsheets. Access is okay for lightweight stuff, beyond that MySQL is free and good, but quite geeky to start with.
>
> 'Use visualisations for research/exploring data, it can help you see a story in the data, where to target your research/legwork. Often it might be nothing significant, but you might need to write about it, explain it, and sometimes it will be the gem of the story.
>
> 'Displaying for the web is often quite tricky compared with the stuff above. Good data doesn't necessarily lead to something cool which can be displayed. Logarithmic scales are often required for displaying data in a meaningful way, but they can't really be used for public facing stuff as they are generally considered too complex outside academia.'

The Wall Street Journal Guide to Information Graphics (2010) offers a wealth of tips on elements to consider and mistakes to avoid in both visualisation and data research. Here is just a selection:

- 'Choose the best data series to illustrate your point, e.g. market share vs total revenue.
- 'Filter and simplify the data to deliver the essence of the data to your intended audience.
- 'Make numerical adjustments to the raw data to enhance your point, e.g. absolute values vs percentage change.
- 'Choose the appropriate chart settings, e.g. scale, y-axis increments and baseline.
- 'If the raw data is insufficient to tell the story, do not add decorative elements. Instead, research additional sources and adjust data to stay on point.
- 'Data is only as good as its source. Getting data from reputable and impartial sources is critical. For example, data should be benchmarked against a third party to avoid bias and add credibility.
- 'In the research stage, a bigger data set allows more in-depth analysis. In the edit phase, it is important to assess whether all your extra information buries the main point of the story or enhances [it].'

Visualising large amounts of text

If you are working with text rather than numbers there are ways to visualise that as well. *Word clouds*, for instance, show which words are used most often in a particular document (such as a speech, bill or manifesto) or data stream (such as an RSS feed of what people are saying on Twitter or blogs). This can be particularly useful in drawing out the themes of a politician's speech, for example, or the reaction from people online to a particular event. They can also be used to draw comparisons – word clouds have been used in the past to compare the inaugural speeches of Barack Obama with those of Bush and Clinton; and to compare the 2010 UK election manifestos of the Labour and Conservative parties. The *tag cloud* is similar to the word cloud, but typically allows you to click on an individual tag (word or phrase) to see where it has been used.

There are other forms for word visualisation too, particularly concerning showing relationships between words – when they occur together, or how often. The terminology varies: visualisation

tool ManyEyes, for example, calls these *word trees* and *phrase nets* but other tools will have different names.

Visualisation tools

So, if you want to visualise some data or text, how do you do it? Thankfully there are now dozens of free and cheap pieces of software that you can use to quickly turn your tables into charts, graphs and clouds.

The best-known tool for creating word clouds is Wordle (www.wordle.net). Simply paste a block of text into the site, or the address of an RSS feed, and the site will generate a word cloud whose fonts and colours you can change to your preferences. Similar tools include Tagxedo (tagxedo.com/app.html), which allows you to put your word cloud into a particular shape.

ManyEyes also allows you to create word clouds and tag clouds, as well as word trees and phrase nets that allow you to see common phrases. But it is perhaps most useful in allowing you to easily create scattergrams, bar charts, bubble charts and other forms. The site also contains a raft of existing data that you can play with to get a feel for the site. Similar tools that allow access to other data include Factual, Swivel, Socrata and Verifiable.com. Google Fusion Tables is particularly useful if you want to collaborate on tables of data, as well as offering visualisation options. A more powerful service is offered by Dabble DB, which allows you to create, share and collaborate on online databases, as well as visualise them.

More general visualisation tools include Widgenie, iCharts, ChartTool and ChartGo. FusionCharts is a piece of visualisation software with a Google Gadget service that publishers may find useful.

If you want more control over your visualisation – or want it to update dynamically when the source information is updated, Google Chart Tools is worth exploring. This requires some technical knowledge, but there is a lot of guidance and help on the site to get you started quickly.

Tableau Public is a piece of free software you can download (tableausoftware.com/public) with some powerful visualisation options. You will also find visualisation options on spreadsheet applications such as Excel or the free Google Docs spreadsheet service. These are worth exploring as a way to quickly generate charts from your data on the fly.

For visualising maps, Google Maps is an obvious place to start. Beginners can add data manually or use a tool like Map A List to publish their spreadsheet to Google Maps or Google Earth. Also worth exploring is the open source option Open Street Map. Spreadsheets can also be converted to KML format, which opens up various possibilities related to mapping data that you should explore further if you want to develop skills in this area. More broadly, there is a whole school of geovisualisation and geographic information systems (GIS) that you might want to read up on.

Publishing your visualisation

There will come a point when you've visualised your data and need to publish it somehow. The simplest way to do this is to take an image (screengrab) of the chart or graph. This can be done with a web-based screencapture tool like Kwout, a free desktop application like Skitch or Jing, or by simply using the 'Print Screen' button on a PC keyboard (cmd+shift+3 on a Mac) and pasting the screengrab into a graphics package such as Photoshop.

The advantage of using a screengrab is that the image can be easily distributed on social networks, image-sharing websites (such as Flickr), and blogs – driving traffic to the page on your site where it is explained.

If you are more technically minded, you can instead choose to embed your chart or graph. Many visualisation tools will give you a piece of code which you can copy and paste into the

HTML of an article or blog post in the place you wish to display it (this will not work on most third-party blog hosting services, such as WordPress.com). One particular advantage of this approach is that the visualisation can update itself if the source data is updated.

Alternatively, an understanding of Javascript can allow you to build 'progressively enhanced' charts which allow users to access the original data or see what happens when it is changed.

Showing your raw data

It is generally a good idea to give users access to your raw data alongside its visualisation. This not only allows them to check it against your visualisation but adds insights you may not otherwise gain. It is relatively straightforward to publish a spreadsheet online using Google Docs (see Closer look: How to publish a spreadsheet online, below).

Mashing data

Wikipedia defines a mashup as 'a web page or application that uses or combines data or functionality from two or many more external sources to create a new service.' Those sources may be

Closer look How to publish a spreadsheet online

Google Docs is a free website which allows you to create and share documents. You can share them via email, by publishing them as a webpage, or by embedding your document in another webpage, such as a blog post. This is how you share a spreadsheet:

1. Open your spreadsheet in Google Docs. You can upload a spreadsheet into Google Docs if you've created it elsewhere. There is a size limit, however, so if you are told the file is too big try removing unnecessary sheets or columns.
2. Look for the 'Share' button (currently in the top right corner) and click on it.
3. A drop-down menu should appear. Click on 'Publish as a web page'.
4. A new window should appear asking which sheets you want to publish. Select the sheet you want to publish and click 'Start publishing' (you should also make sure '*Automatically republish when changes are made*' is ticked if you want the public version of the spreadsheet to update with any data you add).
5. Now the bottom half of that window – '*Get a link to the published data*' – should become active. In the bottom box should be a web address where you can now see the public version of your spreadsheet. If you want to share that, copy the address and test that it works in a web browser. You can now link to it from any webpage.
6. Alternatively, you can embed your spreadsheet – or part of it – in another webpage. To do this, click on the first drop-down menu in this area – it will currently say '*Web page*' – and change it to '*HTML to embed in a page*'. Now the bottom box on this window should show some HTML that begins with <iframe . . . Copy this and paste it into the HTML of a webpage or blog post to embed it (embedding may not work on some third-party blog hosting services, such as WordPress.com).
7. If you want to embed just part of a spreadsheet, in the box that currently says '*All cells*' type the range of cells you wish to show. For example, typing A1:G10 will select all the cells in your spreadsheet from A1 (the first row of column A) to G10 (the 10th row of column G). Once again, the HTML below will change so that it only displays that section of your spreadsheet.

online spreadsheets or tables; maps; RSS feeds (which could be anything from Twitter tweets, blog posts or news articles to images, video, audio or search results); or anything else which is structured enough to 'match' against another source.

This 'match' is typically what makes a mashup. It might be matching a city mentioned in a news article against the same city in a map; or it may be matching the name of an author with that same name in the tags of a photo; or matching the search results for 'earthquake' from a number of different sources. The results can be useful to you as a journalist, to the user, or both.

Why make a mashup?

Mashups can be particularly useful in providing live coverage of a particular event or ongoing issue – mashing images from a protest march, for example, against a map. Creating a mashup online is not too dissimilar from how, in broadcast journalism, you might set up cameras at key points around a physical location in anticipation of an event from which you will later 'pull' live feeds. In a mashup you are effectively doing exactly the same thing – but in a virtual space rather than a physical one. So, instead of setting up a feed at the corner of an important junction, you might decide to pull a feed from Flickr of any images that are tagged with the words 'protest' and 'anti-fascist'.

Some web developers have built entire sites that are mashups. Twazzup (www.twazzup.com), for example, will show you a mix of Twitter tweets, images from Flickr, news updates and websites – all based on the search term you enter. Friendfeed (friendfeed.com) pulls in data that you and your social circle post to a range of social networking sites, and displays them in one place.

Mashups also provide a different way for users to interact with content – either by choosing how to navigate (for instance by using a map), or by inviting them to input something (for instance a search term, or selecting a point on a slider). The Super Tuesday YouTube/Google Maps mashup, for example, provided an at-a-glance overview of what election-related videos were being uploaded where across the USA (Pegg, 2008).

Finally, mashups offer an opportunity for juxtaposing different datasets to provide fresh, sometimes ongoing, insights. The MySociety/Channel 4 project Mapumental, for example, combines house price data with travel information and data on the 'scenicness' of different locations to provide an interactive map of a location which the user can interrogate based on their individual preferences.

Mashup tools

Like so many aspects of online journalism, the ease with which you can create a mashup has increased significantly in recent years. An increase in the number and power of online tools, combined with the increasing 'mashability' of websites and data, means that journalists can now create a basic mashup through the simple procedures of drag-and-drop or copy-and-paste.

A simple RSS mashup, which combines the feeds from a number of different sources into one, for example, can now be created using an RSS aggregator such as xFruits (www.xfruits.com) or Jumbra (www.jumbra.com). Likewise, you can mix two maps together using the website MapTube (maptube.org), which also contains a number of maps for you to play with.

And if you want to mix two sources of data into one visualisation the site DataMasher (www.datamasher.org) will let you do that – although you will have to make do with the US data that the site provides. Google Public Data Explorer (www.google.com/publicdata) is a similar tool which allows you to play with global data.

However, perhaps the most useful tool for news mashups is Yahoo! Pipes (pipes.yahoo.com).

Yahoo! Pipes allows you to choose a source of data – it might be an RSS feed, an online spreadsheet or something that the user will input – and do a variety of things with it. Here are just some of the basic things you might do:

- Add it to other sources
- Combine it with other sources – for instance, matching images to text
- Filter it
- Count it
- Annotate it
- Translate it
- Create a gallery from the results
- Place results on a map.

You could write a whole book on how to use Yahoo! Pipes – indeed, people have – so we will not cover the practicalities of using all of those features here. There are also dozens of websites and help files devoted to the site. However, below is a short tutorial to introduce you to the website and how it works – this is a good way to understand how basic mashups work, and how easily they can be created.

Mashups and APIs

Although there are a number of easy-to-use mashup creators listed above, really impressive mashups tend to be written by people with a knowledge of programming languages, and who use APIs. APIs (Application Programming Interfaces) allow websites to interact with other websites. The launch of the Google Maps API in 2005, for example, has been described as the 'tipping point' in mashup history (Duvander, 2008) as it allowed web developers to 'mash' countless other sources of data with maps. Since then it has become commonplace for new websites, particularly in the social media arena, to launch their own APIs in order to allow web developers to do interesting things with their feeds and data – not just mashups, but applications and services too. The API Playground (apiplayground.org) is one website which was built to allow journalists to play with a selection of APIs in order to see what they do.

If you want to develop a particularly ambitious mashup, it is likely that you will need to teach yourself some programming skills, and familiarise yourself with some APIs (the APIs of Twitter, Google Maps and Flickr are good places to start).

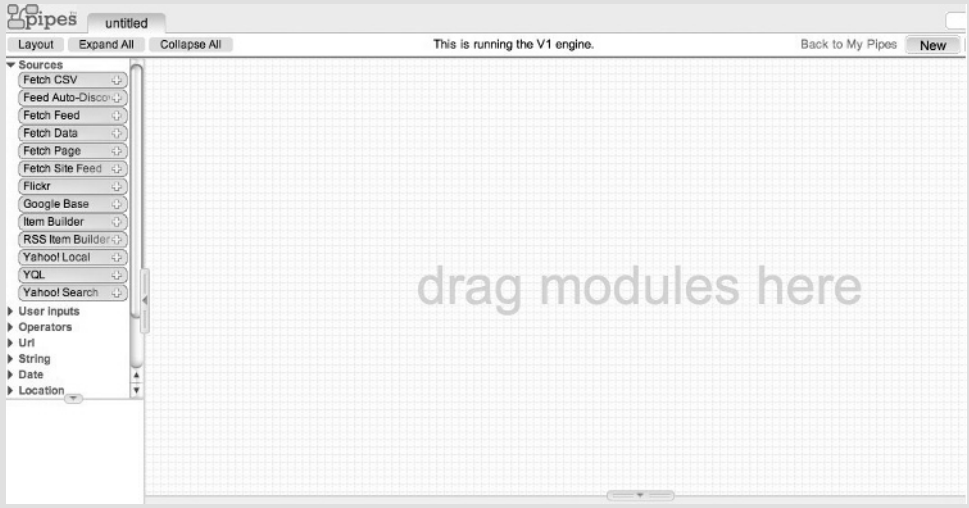
Closer look Using Yahoo! Pipes to create a sports news widget

Signing up

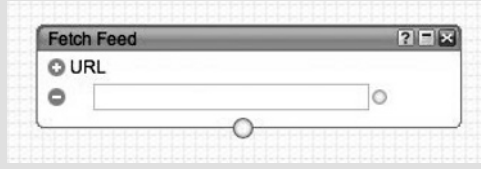
First you'll need to go to pipes.yahoo.com and register. If you already have a Yahoo! or Flickr account you may be able to use that.

Aggregating feeds into one

1. Log on to Yahoo! Pipes and click on *Create a pipe*. You should be presented with a 'graph'-style page.



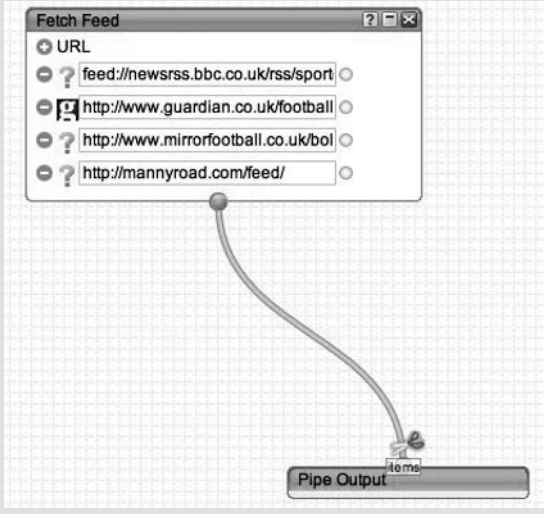
2. On the left column are a number of 'buttons' called modules. These are arranged within different categories, the first category being *Sources*. In the *Sources* category there should be a module called *Fetch Feed*. Click and drag this on to the graphed area.



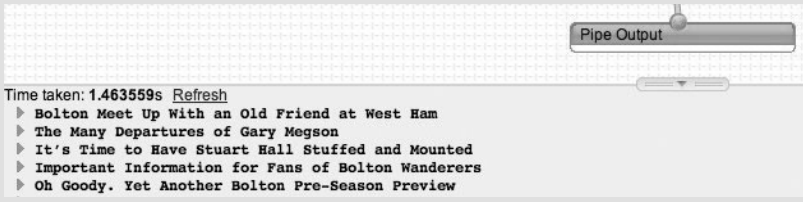
3. Find the URL of an RSS feed you want to aggregate – in this example we'll use the RSS feed for BBC Sport's coverage of Bolton Wanderers (right-click on the RSS icon and copy the link – it should look something like feed://newsrss.bbc.co.uk/rss/sportonline_uk_edition/football/teams/b/bolton_wanderers/rss.xml). Paste the URL into the *Fetch Feed* module input box.

4. To add extra feeds, click on the plus (+) icon next to the URL and further input boxes will appear. Find the RSS feeds for Bolton Wanderers coverage on other news websites and blogs, and add these in each new box.

5. Finally, you need to connect the *Fetch Feed* module to the *Pipe Output*. To do this, click on the circle at the bottom of the *Fetch Feed* module and drag it to the circle at the top of *Pipe Output*. You should now see a pipe appear connecting the two.



6. Click on *Pipe Output* to see the results at the bottom of the screen.



7. That's it. Click *Save* (top right), give the pipe a name, then click *Run Pipe* . . . at the top of the screen. If the results are displayed as images click *List* to see the text version.

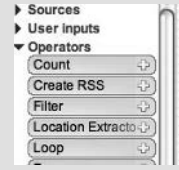
8. Finally, you need to be able to publish the results elsewhere. To do this you need the new RSS feed that has been generated. Along the top of the results you will see various options. Click on the orange *Get as RSS* button to get the RSS feed for the output of this pipe. Copy the URL of that feed

to use in an RSS widget in your blog (see How to blog, Chapter 6, for more on widgets). You can also click *Get as a Badge* to get some HTML that can be put on any website to display results more attractively.

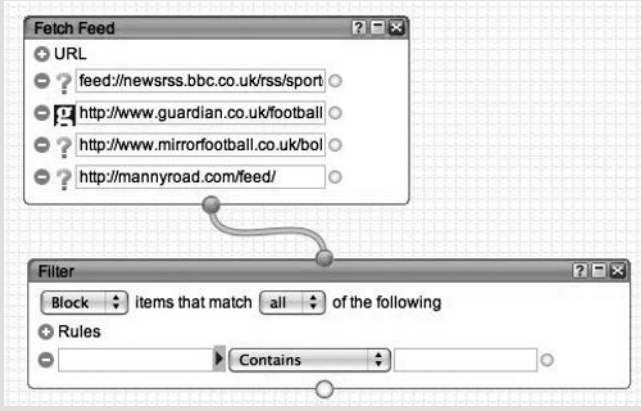
Notes: When you use more than one feed, Yahoo! Pipes will 'cluster' them together by feed rather than by date. To order the results by date, use the *Sort* module under the *Operators* category, connect it to *Fetch Feed*, and *sort by item.pubDate* in *descending* order.

If you want to aggregate feeds after filtering, etc. you can use the *Union* module under *Operators* category.

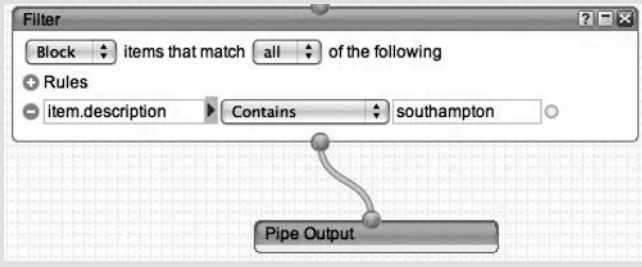
Filtering feeds



1. Follow steps 1–4 for *Aggregating feeds*, above.
2. On the left column in Yahoo! Pipes are buttons, called modules. These are arranged within different categories. Expand the *Operators* category. There should be a module called *Filter*. Click and drag this on to the graphed area.
3. You need to connect the *Fetch Feed* module to the *Filter* module. To do this, click on the circle at the bottom of the *Fetch Feed* module and drag it to the circle at the top of *Filter*. You should now see a pipe appear connecting the two.



4. Using the settings in the Filter module you can choose to filter the aggregated feed by blocking items (articles) with certain words, or only allowing items with certain words to come through. You will need to choose from the drop-down menu which field (e.g. 'title' or 'category') you want to be the subject of the filter.
5. Finally, you need to connect the *Filter* module to the *Pipe Output*. To do this, click on the circle at the bottom of the *Filter* module and drag it to the circle at the top of *Pipe Output*. You should now see a pipe appear connecting the two.



6. Follow steps 6–8 for Aggregating feeds, above, to finish.

All of the above screenshots were created using YahooPipes, http://pipes.yahoo.com/pipes

Note: You can use the *Unique* module instead to filter out several versions of the same post (e.g. when you're using feeds from search results on different engines).

You can see the pipe created above by going to http://bit.ly/ojbkpipeseg – click *Clone* to add it to your own pipes where you can see how it works and adapt it for different purposes.

Closer look Taking it further – resources on programming

If you want to explore programming in more depth there are resources available specifically for journalists. Hackety Hack is a tutorial for learning Ruby on Rails which comes with its own programming environment. Scraperwiki is a website that provides an environment for writing programming scripts in Python so that, again, you don't have to deal with the difficulties of setting one up on your own computer. The site also includes tutorials. More broadly, Help.HacksHackers.com is an online community where journalists and technologists come together to solve problems.

It is always worth attending some events to meet programmers – try searching a site like MeetUp for groups near you interested in a particular programming language.

Closer look Anatomy of a tweet

Plate 6 shows the code behind a simple Twitter update. It includes information about the author, their location, whether the update was a reply to someone else, what time and where it was created, and lots more besides. Each of these values can be used by a mashup in various ways – for example, you might match the author of this tweet with the author of a blog or image; you might match its time against other things being published at that moment; or you might use their location to plot this update on a map.

While the code can be intimidating, you do not need to understand programming in order to be able to do things with it.

Summary

There is a quiet movement taking place in journalism: a generation of journalists who grew up with computers are using them to help do better journalism.

Charles Arthur, a *Guardian* technology journalist, says:

'When I was really young, I read a book about computers which made the point – rather effectively – that if you found yourself doing the same process again and again, you should hand it over to a computer. That became a rule for me: never do some task more than once if you can possibly get a computer to do it.

'I got into data journalism because I also did statistics – and that taught me that people are notoriously bad at understanding data. Visualisation and simplification and exposition are key to helping people understand.

'So data journalism is a compound of all those things: determination to make the computer do the slog, confidence that I can program it to, and the desire to tell the story that the data is holding and hiding.'

Mary Hamilton from the *Eastern Daily Press* says: 'I love coding when it works well, I love that moment of unlocking something or creating something new and useful. I find it oddly exciting, which is probably why I carried on doing it after the first couple of times.'

The Times' Jonathan Richards sees the flood of information online as presenting 'an amazing opportunity' for journalists, but also a challenge:

'How on earth does one keep up with; make sense of it? You could go about it in the traditional way, fossicking in individual sites, but much of the journalistic value in this outpouring, it seems, comes in the form of aggregation: in processing large amounts of data, distilling them, and exploring them for patterns. To do that – unless you're superhuman, or have a small army of volunteers – you need the help of a computer.

'I "got into" data journalism because I find this mix exciting. It appeals to the traditional journalistic instinct, but also calls for a new skill which, once harnessed, dramatically expands the realm of "stories I could possibly investigate . . ."'

Mark Donoghue makes three connections between journalism and computer science, and how they are able to help each other:

'Journalism taught me how to ask questions. Computer Science taught me the importance of asking the right question.

'Journalism taught me how to communicate. Computer Science taught me how to think.

'Journalism taught me how to identify problems. Computer Science taught me how to solve problems.'

As journalists we are living in an era of information overload. Data journalism gives us the skills to find the inaccessible information that our competitors might overlook; the understanding to find the stories and people behind the numbers; and the ability to make those stories understood by users.

These are all traditional skills – with the difference that we have new, powerful tools to use in their practice.

In addition, data journalism offers a way to engage users with the news in ways that print and broadcast journalism could never do. We can present it in ways that allow different people to look at it in different ways. We can create mashups that update the picture in real time. And we can give users access to all of our data and invite them to help shed light on it.

It is a newsroom without walls: there will be better programmers out there, better designers, people with a better understanding of statistics, or who can access information that we can't. All of these people will be potential colleagues in the pursuit of journalism – and joining online communities of support will be as important in your development as reading books on the subject matter. Your role could be as simple as finding some information and raising a question, as *The Guardian* datablog does – or mashing up one source of data with another to provide useful insights. The story – as ever, with online journalism – never ends.

This chapter has touched on disciplines as diverse as statistics, advanced search techniques, information design, research methodology and programming. Knowing a little bit about all of these areas will be useful – but you should continue to read more about them as you progress in your experiments.

And the key thing *is to experiment* – see what data you can find, and what insights you can find within it. Play with visualisation tools and compare the results. Mix different sources of

data together to see what happens. And think how this might be useful as you pursue stories and try to find ways to help users understand complex issues. Be curious. Be creative. The chances are that the results will stick out a mile from 99 per cent of everything else on the subject – and what's more: you'll enjoy it.

Activities

1. Submit a Freedom of Information (FOI) request to your local authority, NHS trust or police force using the website TheyWorkForYou.com. FOIs are best for requesting documents or statistics. You could, for example, ask how much they spent on PR in the past three years, or to see any policy documents relating to dealing with complaints. Use WhatDoTheyKnow to see what other FOI requests have been submitted to that body – and the responses. This will help you formulate your own.
2. Create a simple data mashup with Google Public Data Explorer. Pick any set of data and then select a number of elements to compare (for example, different countries).
3. Use the advanced search techniques outlined in this chapter to find a useful spreadsheet on your local council website.
4. Look at a piece of research that has been released recently by the government or a pressure group. How was the information gathered? How big – and how representative – was the sample? How long a period did the research cover? If any of that information is missing, try to get hold of it – and if they won't provide it, ask why.
5. Put a recent speech by a politician into Wordle and create a word cloud. What does it imply about the politician's priorities, or the point they're trying to make?

Further reading

Blastland and Dilnot (2007) *The Tiger That Isn't*, UK: Profile Books.
Brooke, H. (2010) *The Silent State*, UK: William Heinemann.
Fry, B. (2008) *Visualizing Data*, USA: O'Reilly.
Huff, D. (1954) *How to Lie With Statistics*, UK: Penguin Business.
Meyer, P. (2001) *Precision Journalism*, USA: Rowman & Littlefield. The previous edition is available free online at: www.unc.edu/~pmeyer/book/.
Wong, D. M. (2010) *The Wall Street Journal Guide to Information Graphics*, USA: W.W. Norton & Co.

Blogs

OUseful.Info *blog.ouseful.info*

Information is Beautiful *www.informationisbeautiful.net*

Flowing Data *FlowingData.com*

Google Maps Mania *http://googlemapsmania.blogspot.com/*

Tripwire Magazine: 75+ Tools for Visualizing your Data, CSS, Flash, jQuery, PHP, *www.tripwiremagazine.com/2009/12/70-tools-for-visualizing-your-data-css-flash-jquery-php.html*