# Promulgating the Web's calorie controlled diet – *web design, environmental impact and the much ignored ecological efficiency of the Internet*

Whilst catching up on a long-overdue chore as I recover from the flu, I muse on the role of "design" within the driving energy and resource trends of information systems, and how we measure such ethereal trends in order to define a process for change.

Banbury, Monday 23rd May, 2011.

I'm feeling pretty awful; for the last few days I've been laid low with a bug that just won't go away. As I sit, trying to find something to do, it occurs to me that I could catch-up on some of the really tedious, dead-head chores that I've been putting off for a while. If I feel so awful, how more awful can it be to do those things that I never feel like doing in any case? I begin by trying to write a long-overdue beginner's guide to the Linux [com-mand line inter-face](1) – I get as far as designing a rather enter-taining logo before realising this requires far too much brain power for my current state of mind!

Then I remember the ["design statement"](2) for the Free Range Activism Web Site. That requires measuring lots of web pages to demonstrate, statistically, why the design system for the FRAW site is, ecologically, better than mainstream design methods. Hmmn, yeah, downloading lots of web pages, categorising their component parts and then spreadsheeting the results for later analysis. OK, as occupations go it's the digital equivalent of watching paint dry, but right now I feel that I can do that!
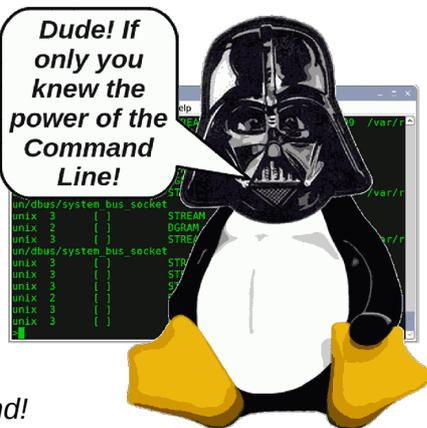
["What is not measured is not man-aged!"](3) My time as an environmental consultant taught me that about many aspects of industrial waste and pollution; and the more I get involved in the ecological implications of information and communications technologies (ICTs), the more I find that same principle here too. That whole approach is, in fact, symptomatic of the present problems with data bloat on-line. Not simply good old-fashioned [software bloat](4) – the historic and inexorable bloat of the computer programs and operating systems that surround our everyday lives. I'm talking about the splurge of additional download content that now plagues the everyday use of the World-wide web. Driven by the [needs of advertisers](5), we see some of the most popular sites on the Web emblazoned with animated technicolour images, trying to peddle everything from the latest must-have gadget to the most unpopular political parties. At the same time, due to the need to manage all this extra data, and the extra dynamic functionality all this splurge of content requires, the tools we use to access the Web are sympathetically [bloating too](6) (see also [this](#))

Let's face it, ICTs and the whole technological storm of the on-line world is administrated by tech-head geeks; generalising, many techies are about as ecologically minded about the ecological footprint of ICTs as the BBC's *[Top Gear](#) team*[7] are on the ecological impacts of motoring. If the geeks don't measure the extent of the ecological impacts of ICTs generally, what hope is there of address the bloat of on-line services?

A while ago, whilst browsing in the British Library, I came across a paper that, at a more general level, hit this issue right on the head[8] –

> The need for the management of IT results has not been universally accepted or understood. While solutions have been sought to the axiom "what gets measured gets managed", a single set of guiding measures has never been developed... The 'IT productivity paradox' is that over the years, despite the large investments that organisations have made in IT, they continue to question its value because of the lack of empirical evidence that demonstrates that the IT organisation operates effectively and efficiently.

We measure much of the bloat of the on-line world without realising it: When you pay for extra giga-bytes of broadband capacity; when you're hard disk drive fills up and you have to install another; when you're web browsing software can't cope with the latest standards and you have to upgrade to keep accessing the latest content; or when you end up burning lots of DVD discs in order to back-up the data from your computer – *you're paying for bloat!*

Some would say that this is simply for the "force of progress". Of course, it's all a matter of perspective: Thirty years ago my first computer[9] had a *kilobyte* of memory – my present work machine has 2 million times more!; it had a *3.25MHz processor* – my present work machine shifts data at well over 5,500 times faster than that; my first modem[10] ran at *300 bits/second* – my router tells me that it's currently able to run 20,000 times faster than that; my first PC hard disk had a total capacity of *20 mega-bytes* – stringing together all the machines and storage devices on my network, I now have about 300 million times more capacity than that! Today we're encouraged by "tech. culture" to consume the latest technology and junk the perfectly serviceable device[11]. From the perspective of someone who learnt their skills when there were hard limits to our everyday use of technology, all the kipple[12] that pervades our use of technology today seems so unnecessary.

The IT industry has been able to grow significantly over of the last thirty years due, in a large part, to the increasing power and processing capacity of the equipment involved. It's really easy to grow your industry when the tools of your trade double their power and halve their utilisation costs every 18 months or so[13]. You don't have to work at being more productive – *getting a greater output from your existing body of production resource*s – when the power of those systems is growing exponentially.

As I see it this is the root of the "IT productivity paradox"; if you don't have to work hard to increase your productive capacity you're going to produce rubbish – you could produce any old rubbish because the rising power, capacity and functionality of technology is able to make-up for your lack of attention to the efficiency of design.

This, of course, is where the problem of bloat arises. If computers get more powerful you have no incentive to improve; and as for web bloat, if network speeds are rising then you have no need to get more creative with your available resources. For example, when mobile phone operators paid billions for their licenses, they had an incentive to get as much capacity out of their networks as possible – and through new transmission protocols they achieved a higher capacity[14] than initially expected. In contrast, the designers of web systems have no such pressures on the quality and efficiency of their work. By definition, those with the money to buy goods and services online are also likely to have the money to buy the broadband and high-powered computer equipment required to assimilate the web bloat without a problem. Consequently it doesn't matter to the designer's business whether or not those with dial-up access, or who can't afford the computer hardware and software to handle the bloat, are marginalised by such digital profligacy – *provided that those with money are OK, the business model works*.

That is, of course, a very short-sighted way of looking at bloat. Irrespective of whether we access the web or not, *we're all paying for bloat*. Not just in the extra money we have to fork-out to download and manipulate all that data – *we're paying for it ecologically*. Transferring greater quantities of data requires us to buy higher-capacity hardware, but that's also using up the finite stocks of rare metals, and generating toxic waste streams[15] as a result of the inappropriate management of e-waste[16]. Making all that equipment also uses a large quantity of energy, as does running it – which also contributes to our depletion of finite energy resources, the production of pollution and climate change[17].

Today the internet and its associated gadgets and hardware are using about 5% of global electricity production[18], and producing as much carbon as the airline industry[19]. Recent studies commissioned by the European Union[20] estimate the total electricity drain of ICT at about 8% of EU electricity generation, equivalent to 98 mega-tonnes (or 1.9%) of EU carbon emissions. This is projected to rise to 10.5% of electricity production in 2020 (the figures for the whole EU are likely to be roughly accurate for the UK individually). ***Why are these impacts growing?; in part because those who design web standards, write the programs to implement them and design the on-line services, do not measure and study the impacts of their work!***

Ultimately it requires the environmental campaign groups (who, if you look at their web sites, also seem to be besotted by the whole *web-techno-bloat* gig) to get their heads around this, take this issue on and make it a talking point. It's a simple, easy to target campaigning gift, if they choose to see it that way.

I can't decide what happens across the Internet, but I can do something about the small part of it that I organise. When the Free Range Network began the redesign of our web site last year we perused this issue, and found that it was possible to make a major difference to the impact of running a web site. It's not so much a technological issue, or the type of content you create, *it's all about underline design*; you have to deliberately set out to create a site that uses the least possible resources. Not only does that reduce the amount of energy it takes to load our pages, but we also freed-up 20% of the web storage for further information as a result of the re-design (although that's a purely theoretical saving – we have since filled it with new information we didn't previously have the space for!).

Measuring how much "better" that approach is to more conventional site designs is not a straightforward issue. How do you measure the difference?; what indicators do you seek to quantify? No matter how you decide to measure the difference, to contrast this new design with other web sites we need to measure a lot of other web pages; and to interpret what data this produces it's necessary to understand how the web systems works (see the box on the next page for an explanation of web design).

Let's begin with a simple premise –

> *the quantity of data transacted*
> *is proportional to impact*

That's a straightforward idea to characterise:

- A static web page that's held in the magnetism of a spinning hard disk until it is requested requires a different level of energy supply to a database-driven or dynamic web site that must be actively maintained by the server;
- A site that uses more dynamic code to structure and manage the page will have a greater impact than a site that uses little;
- A site that transacts comparatively few files to serve a single page will have a lower impact than a site that serves many files per page;
- A site that optimises the design of the pages, to convey as much "visible" information using the least "invisible" formatting information and script code, will have a lower impact.

This is not an absolute scale we're talking about. Obviously the impact of a small site is going to be very different to that of a large, highly-used site. The issue isn't so much the absolute level of impact, but the comparative impact within the class of site involved. In effect then, it's a competition between similar classes of application, not an absolute measure across all sites.

Without access to the highly sensitive commercial information from a large number of web servers – such as levels of traffic, system utilisation and power consumption – constructing such measures remotely

for individual sites is also difficult. We can, however, contrast different sites in order to generalise about the relative impacts between different sites within a similar class. By downloading a representative set of web pages and analysing their content and structure we can compare, using the list above, the relative impact of each site. This presents two problems – selecting the sites, and selecting a sample of pages from each site.

Selecting sites is relatively easy. We can identify sites that fall under a broad class of subject or application, and then pick a number of sites that are representative of that classification. For the purposes of this study I've selected four categories[21], with four sites per category, and the FRAW site:

- Mainstream media
  - *BBC* [#]
  - *The Independent* [#]
  - *Guardian/Observer* [#]
  - *The Telegraph* [#]
- Alternative media
  - *Alternet* [+]
  - *CorporateWatch* [+]
  - *Ecologist* [+]
  - *Media Lens* [+]
- Campaign groups
  - *FoE* [+]
  - *Greenpeace* [+]
  - *RSPB* [+]
  - *WWF* [+]
- Political Parties
  - *Conservatives*
  - *Greens* [+]
  - *Labour* [+]
  - *Lib-Dems* [+]
- FRAW Site

Selecting which files from these to sample is a more difficult issue – how do you select files without creating a bias towards certain types of web resources and not others. The simplest solution turned out to be the most obvious; *let the site self-select the pages*. Many sites have a "most popular" menu – this was used for a number of sites to select the ten most popular files from each site (marked [#] above). For those sites without a "most popular" list I took the first ten items from the "latest" or "news" menus (marked [+] above). The exception to this was the Conservative Party site; their "most popular" list was only 5 item long (due to the cuts??), and so I also took the top 5 menu items too. Finally, for the FRAW site, I took the ten most popular articles identified in the site's usage statistics for April 2011.

Downloading the above with *Firefox*[22], using the "web page, complete" option to save each web page and all its associated files, produced 170 cached web pages. ***In total those 170 web pages comprised nearly 10,000 files, amounting to nearly 160 mega-bytes of data!*** Here, at the first hurdle, we see the bloat issue manifesting itself!

Web design and the quantification of bloat

To understand how to measure bloat you need to look at how web services operate. When the web was first used a "web page" was simply a single file that the browser downloaded from a remote computer, and displayed on your screen using 23. "HTML tags"[23] to format the contents. Today it's a very different system (see the diagram below).

Now, especially for large sites, it's rare to find a truly "static" web site[24] – one where the pages are specific files on a web server. Instead many web sites today are run as large databases. When you request a page the server works out which components it needs to create the page, uses a set of templates[25] to cobble the page together from it's database system, and then sends you the page. The principal difference is that a "static" site is stored, passively, on a hard disk, and so doesn't incur much energy use to maintain it on-line unless it's actually requested by someone. Database driven and/or "dynamic" sites[26] (created by computer programs) require a much greater overhead of computer power, both to keep them available and to serve the page when its requested.

The basic web page is just that – a page containing text and formatting instructions (if you'd like to view the formatting, go to a web page and then select "view source" from your browser menu). The way that other files – such as images, animated images and sounds – are added to the basic page is with links. When your browser downloads the basic page it reads it, makes a list of all the additional files it needs to download and then fetches them too. Consequently a "web page" isn't simply a page, it's a collection of files that are knitted together by the instructions the main page contains.

The design information that a page uses is usually controlled by another file called a *style sheet*[27]– sometimes you have a whole collection of them. The benefit of using style sheets is that a large number of pages can have their design consistently controlled by a single file, which makes maintaining a large web site much easier. In addition to this you also may have one or more *script files*[28], small programs, usually using a language called Javascript[29], which control the operation of active elements on the page (buttons, images, forms, drop-down menus, etc.).
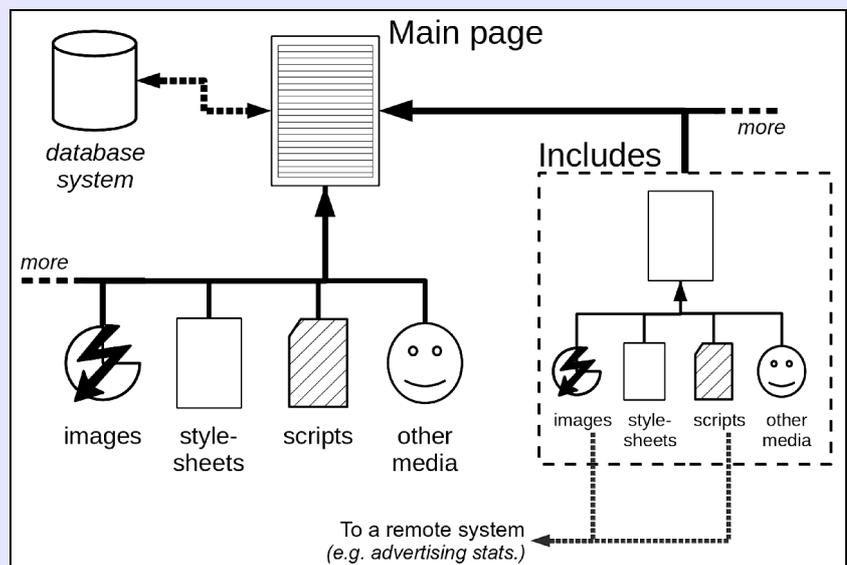
Now things get even more complex. A web page can call on "includes"[30] – in effect the contents of another page, with any

files it requires to be downloaded, is inserted into the main page; alternately an include can be displayed within the main page as a "frame"[31]. Either option can rapidly increase the amount of data downloaded as the include will have its own scripts and style sheets, as well as other media files.

Finally, the need to monitor web usage[32] adds even more to this burden. Advertising takes up a lot of space, and uses a lot of high-quality and animated images. What really slows the loading of pages are links to statistics collection sites. Often this involves downloading a minute image, just one pixel square, in order to clock-up one page view on a log somewhere. Even though it's small, if the site collecting the data is overloaded – because lots of other pages are also polling it to log a download – then it slows the downloading of any web pages that use that service.

OK, let's put all this together really easily within a method to measure web bloat.

Many web browsers, such as *Firefox*, allow you to save the entire web page – usually in a folder which contains all the files required to display the page from your hard disk. This is a very simple way to collect all the data associated with a certain page. By downloading a selected number of pages we can collect a folder-full of representative pages from a range of web sites. Then, by comparing the contents of the folders containing the data for each page, we can measure the attributes that define the page and its design philosophy – and then contrast the differences between sites. If we can collect enough pages, and measure all their associated files, we can then find patterns by comparing sites against one another rather than focussing on the characteristics of a single web site or a selection of pages from a single site.

Main page

database system

more

Includes

more

images   style-sheets   scripts   other media

images   style-sheets   scripts   other media

To a remote system
(e.g. advertising stats.)

Crunching numbers is like measuring string – *how much do you need?* As with any such operation, it's always nice to have lots of data, and many ways to describe the information that the data might contain, but ultimately it's a matter of effort versus the potential reward. Sifting 10,000 files is not a simple thing to do by hand – for that reason I opt to take the simplest approach and write a short computer program to do it for me. Half an hour's coding allows me to complete in a few seconds what might otherwise have taken a day or more to achieve manually. For each of the pages:

◆ Within each folder for the 170 pages the main web page is stored in a subfolder called *page*;

◆ Then the images, flash and media files are moved to a sub-folder called *media*;

◆ Then the style and script files are moved to a sub-folder called *format*;

◆ Any remaining files, which are mostly server-side include files, are moved to a subfolder called *content*;

◆ For each web page the size of the *page*, *media*, *format* and *content* sub-folders, and the number of files they contain, is logged;

◆ Next the total size and number of certain file types is logged for later analysis – the *.css* (style sheet) and *.js* (script) files from the format sub-folders, and the *.gif*, *.png*, *.jpg* and *.swf* files in the media sub-folders;

◆ Finally, using all the accumulated data, I sift the data for all 170 web pages to produce histograms[33] in order to perform some frequency analysis of the page content.

The primary purpose of the program is to sift all the downloaded files according to their type or purpose, log the vital statistics of interest, and finally display/export them in a format I can easily dump straight into a spreadsheet. What I have to do manually (because it's the easiest way of doing it) is load each of the 170 web pages, select and 'copy' the visible text, and then dump it into a text editor to measure the number of "visible" characters that each page contains. I'll explain why later.

First and foremost it must be noted that *I don't have enough data*. If I had a couple of weeks I'd write a very complex spider program, that would run on one of my computers 24-hours a day for a week or two, to comb the Web for hundreds of thousands of pages and perform the analysis automatically. Unfortunately I don't have that amount of time available (not unless someone funds some real research!) and
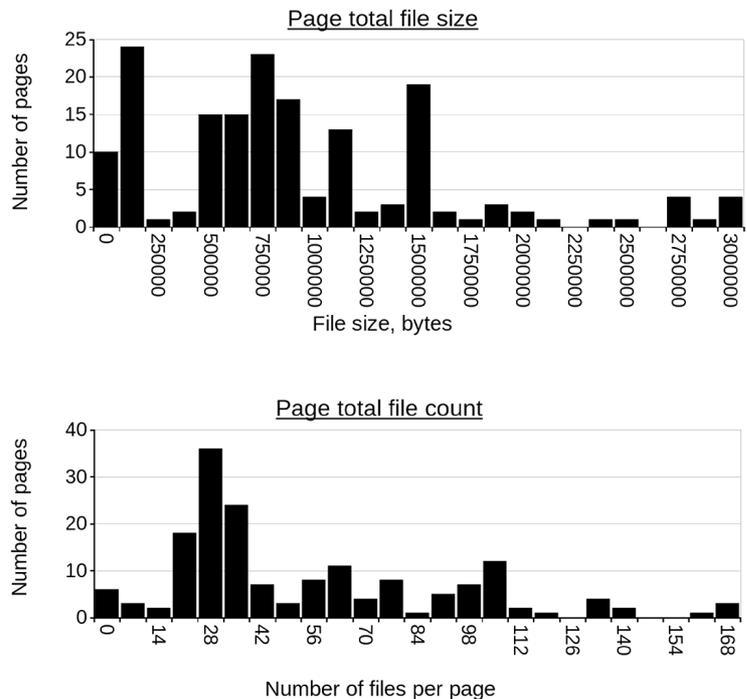

Fig. 1. Analysis of total size/no. of files for each page

so – as noted above – in the struggle between effort and reward in the production of statistics I'll have to settle for what I've got. The important thing to note is that I'm not trying to demonstrate a trend for the *whole web*, but the *difference between* the FRAW site and others.

The problem with my small sample can be seen in the histograms above. These show the frequency of the total file size, and the number of files associated with each page, for the 170 pages downloaded. Whilst you can see that the data clumps around certain regions there is no clear "trend". Sifting hundreds of thousands of pages, rather than 170, would provide a far better analysis that was representative of the web as a whole. However, even from this small sample you can draw some interesting facts. Nearly a fifth of the pages downloaded had 100 or more files associated with them; and whilst only 2% has a size of over 3 mega-bytes, just over a third of all the pages had a size of more than a mega-byte. That said, you might be interested to know that the average file size across the 10 pages from the FRAW site was 169 kilo-bytes (about a sixth of a mega-byte) – on average each page also had just 8 files associated with it *(and interestingly, perhaps due to its popularity, three of those ten pages were past editions of* `ecolonomics`*)*.

Another reason why there is no clear trend is related to the point made earlier. Different web sites have different functions/applications, and this is written as much into the design and scale of the pages as it is in their actual, human-readable content. Unless we differentiate between the applications or organisa-
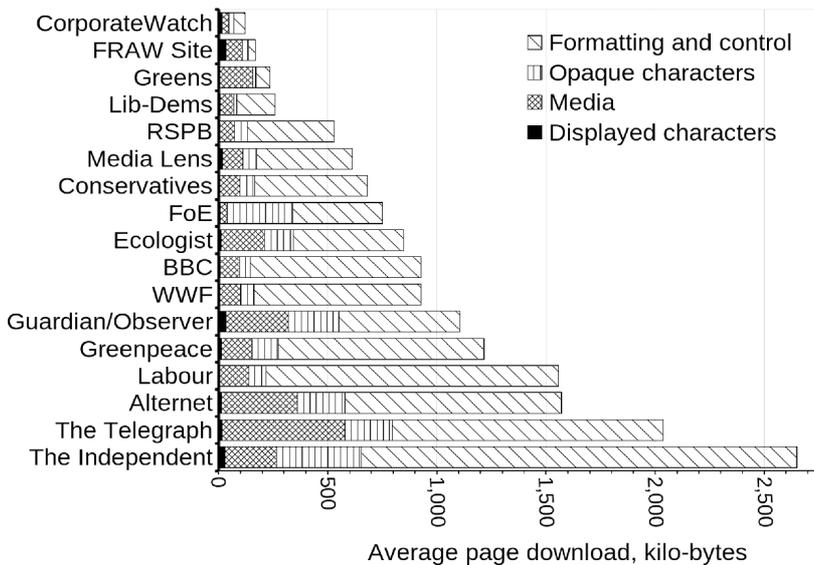
Figure 2 (chart). Vertical axis site labels top to bottom: CorporateWatch, FRAW Site, Greens, Lib-Dems, RSPB, Media Lens, Conservatives, FoE, Ecologist, BBC, WWF, Guardian/Observer, Greenpeace, Labour, Alternet, The Telegraph, The Independent.

Legend:
- ⊠ Formatting and control
- ⊞ Opaque characters
- ⊠ Media
- ■ Displayed characters

Horizontal axis values: 0, 500, 1,000, 1,500, 2,000, 2,500

Average page download, kilo-bytes

Figure 2. Average page size grouped by site

tions involved, and contrast site-to-site, it will be difficult to define any clear trend across all the pages.

You can see this trend more clearly if we calculate the average page size for each site – *as shown in the graph above*. Rather than a uniform distribution, certain sites have much larger files sizes than others. As noted earlier, this is primarily because how web pages are created, the templates, formatting and code they use, are issue of design – *and design policy is an organisational issue*. For example, I think it's a curious that for the average page you might download from *The Independent's* site, you can download two pages comprising roughly the same amount of data from *The Guardian/Observer* site.

In the graph above the total file size is split into its functional components; the "visible" and "opaque" data that make up each page, the media files, and the formatting data. Note that whilst the page/media data increases as file sizes get larger, **what's really ballooning in size is the amount of formatting data** – the style sheets and scripts that are a function of the site's design template. We can see this trend more clearly if we take all the pages and, rather than ordering them by the site, split them into groups (by taking all 170 pages, sorting them by size, and then splitting the sample into equal parts – in this case five parts, or *quintiles*). Again, as size relates to function, we see a trend emerging, as shown in the graph on the right. Rather than showing the total size

as an absolute scale, this illustrates the page composition as a percentage – so that we can compare the relative make-up of each group of pages.

For the smallest pages (the 1st quintile) there's a roughly equal split between page data, media files, style sheet (CSS) and scripting (JS) data. However, as the size of the page grows, the 'formatting' data becomes the dominant component of the page – for the largest pages (the 5th quintile) 70% of the total download is formatting data, *and just over half of all the formatting data is script files!*
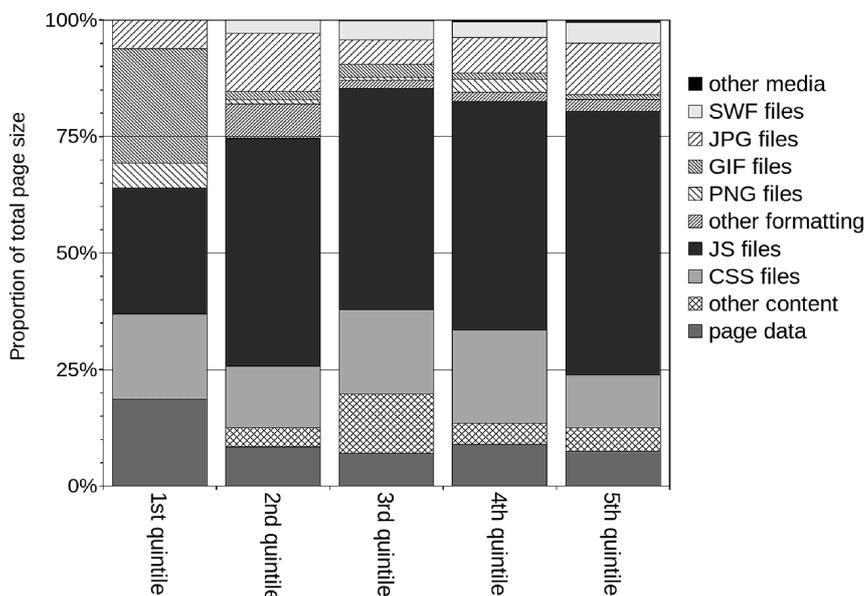
Statistical analysis can be a revealing process, if you enter the process objectively. Perhaps that's the enjoyable pay-off for undertaking what can often be a mind-numbing exercise. When I set out on this analysis I knew that script and style sheets were a large part of web data, but I thought that the media files would be a dominant component of the overall page size. What this analysis shows is that it's the "opaque" components of the page – *the style sheet, scripting and page formatting data* – that are the dominant components of web pages; what you actually see on the screen – *the "visible" data such as text, graphics and animated images* – are only a minor component of the data downloaded. **This is very significant if we're thinking about the ecological footprint of the web – *what's the point of downloading large quantities of data if you can't "see" it?***

We can refine this analysis further by creating histograms of the proportion of the component parts

Figure 3. Comparison of page construction and page size

Figure 3 (chart). Vertical axis: Proportion of total page size, 0% to 100% (0%, 25%, 50%, 75%, 100%). Horizontal axis categories: 1st quintile, 2nd quintile, 3rd quintile, 4th quintile, 5th quintile.

Legend:
- ■ other media
- ☐ SWF files
- ▨ JPG files
- ▩ GIF files
- ◺ PNG files
- ▨ other formatting
- ■ JS files
- ▤ CSS files
- ⊠ other content
- ▦ page data

(page, media and formatting) that make up the 170 web pages. This shows a definitive trend for the functional components of web pages; the average page comprises, at an average ratio of at least 2:1, formatting and control data – and of that formatting and control data, again at a ratio of at least 2:1, it's scripting that makes up the bulk of the data (as illustrated in figure 7, later).

OK, enough generalisation about page construction – *is the FRAW site more "ecological" than any other?* To decide that we need to an indicator – a scale to measure one site against another. Tossing ideas backwards and forwards, trying to encapsulate the many different ways of looking at the ecological performance of how a web page is constructed, I think I've come up with a possible (albeit arbitrary) metric. Remember, *the quantity of data transacted is proportional to impact*. Therefore any measure that reduces the size of the data served, or improves the efficiency of the data served by reducing the data transacted whilst keeping the same "visual" display, will reduce the ecological impact.

As noted above, I measured the quantity of visible text displayed on each page – shown in the *displayed characters* block of figure 2. The proportion of displayed characters to the size of the main page and content files is a good indicator of how "efficient" the HTML mark-up system is; the better the system, the more readable text and the less mark-up text there will be. Dividing the number of displayed characters (let's call it *C*) by the size of the main and included/content pages (call it *P*) produces the *character ratio*, or *C/P* (C divided by P) for short. Next we can say that, generally, having huge amounts of graphics, animated images and other media files in a page (unless absolutely necessary) is a waste of space. The simplest way to express this is as a ratio of the media files (call it *M*) to the total page size (call it *T*) – this creates the *media ratio*, or *M/T*. Finally, let's examine the thorny issue of formatting and control data. Rather than apply any arbitrary rule to weight the issue of "opaque" data, let's treat it just like the media files; we divide the total size of the formatting and control data (call it *F*) by the total page size (call it *T*) – this creates the *format/control ratio*, or *F/T*. Now, with a little fiddling and averaging of the three parameters, what we end up with is a number – let's call it the *bloat indicator* (or *BI*) – which can be described as:

$$ BI = 1 - \frac{\left(1 - \dfrac{C}{P}\right) + \dfrac{M}{T} + \dfrac{F}{T}}{3} $$

We can take the data used to create figure 2 and re-work it to indicate the level of *web bloat* – the result of which is shown in figure 4. For each site a BI
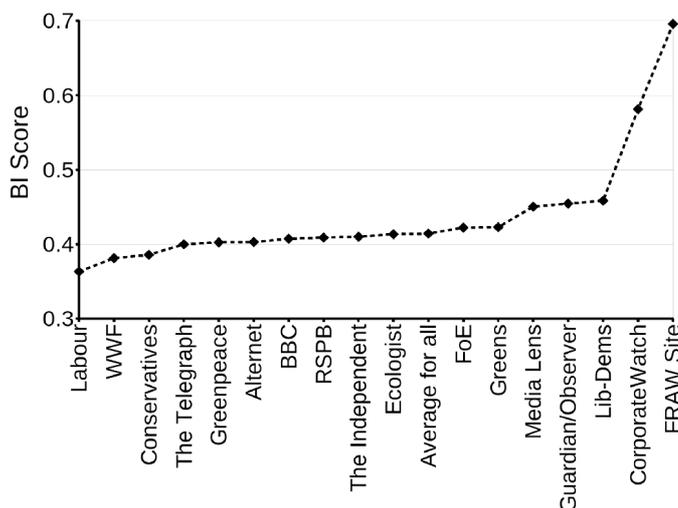
Figures 4. Bloat indicator for 17 sites

value is calculated, and then the results are sorted from the highest to the lowest value (a low score is bad, a high score is good). Note also that the bloat indicator equation averages out the results, meaning that it's the extreme results that are highlighted by the BI figure.

What the results illustrate is a continuum of web design policy, and the effect this has on the transaction of data and thus on impact. This isn't an absolute value, but an average across pages, across a number of sites. Most sites are somewhere around the middle of the graph. That's because they're using the standard design approach that doesn't put great emphasis on the ecological efficiency of design. In contrast the sites falling off the left of the graph, most notably the Labour site, represent sites that, even compared to the mainstream, are very poorly designed or managed. Finally, the sites off at the top right represent the less bloated pages that put a greater proportion of their content into visible information rather than formatting or control... ***and there we find FRAW!***

Of course, *bloat* is only one way of looking at the design issue – and a negative one. Instead of just focussing on the bloat we should also consider the purpose for which web pages are made... *to be seen! T*o counteract the negative pull of the bloat indicator, the last stage of the analysis was to quantify the visibility issue. I covered this earlier in relation to the *character ratio* – the amount of visible text that the web page/page comprises as a proportion of the whole page/content size. For a general visibility indicator we need to include the other major visible component, the media files. To create a value for this, a *visibility ratio* or *visibility indicator* (*VI*), we add together the page content (*C*) and media (*M*) file sizes and divide by the total page size (T):
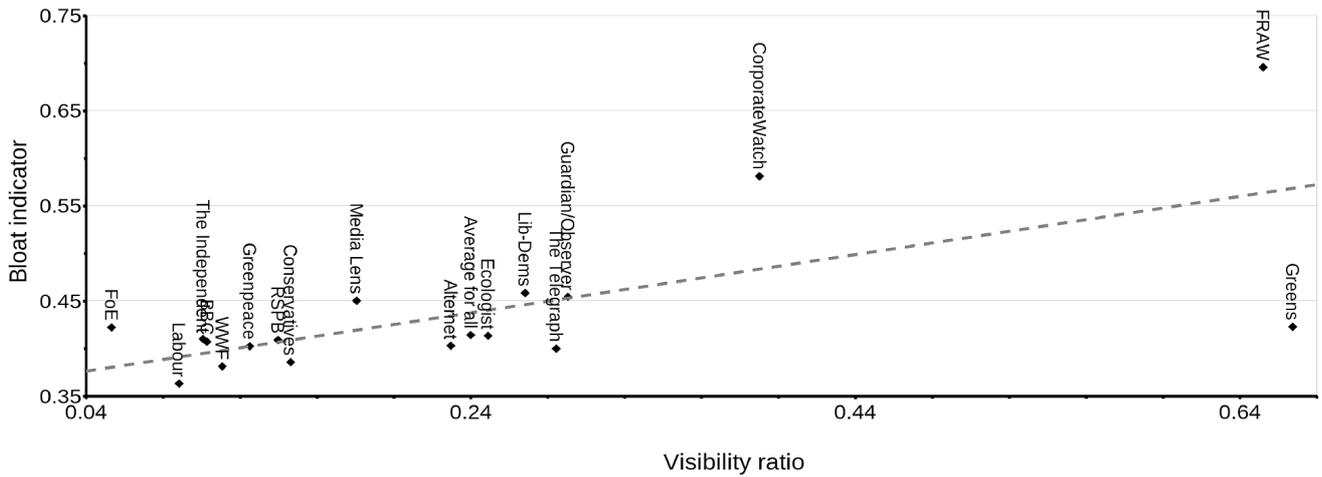
$$ VI = (C + M) \,/\, T $$

Figure 5. Bloat and visibility indicator correlation

Plotting the bloat against the visibility indicator produces an interesting correlation (although that's not surprising as there's a close relationship to the weighting in each value) – as shown in figure 5 above. From the zero value at the bottom left, moving towards the right is good – *it increases visibility*; at the same time move upwards is good – *it decreases bloat*. For that reason the best place to be is the top right, as it represents both high visibility of data and low bloat... **and there we find FRAW!**

What's also notable here is the Green Party site. It gets a high visibility score, higher then FRAW, but also a high bloat score – and for that reason it sits at the bottom right. That's because it uses a lot of graphics for its design and navigation. Consequently, whilst the higher level of graphics and formatting data gives it a high bloat score, the large proportion of graphics also gives it a high visibility. Getting a better result isn't simply a matter of reducing the size of the graphics – that would also reduce their visibility. What they need to do is radically reduce the overhead of page formatting, style sheet and script code; even if they kept the same graphics, that would reduce their bloat score.

Why is FRAW in the "best" location on the graphs? (it's not a fiddle, *honest!*) **It's primarily because of the very low script code and HTML mark-up overhead that our pages embody. That's not surprising...** *they're deliberately designed that way!*

What's interesting is to look at the design of the CorporateWatch web pages. They've got a good bloat score because they're very well designed, minimalist pages. However, they still contain a lot of formatting and control scripts for the navigation system. If they switched to a less intensive navigation scheme they'd probably be on a par with the FRAW site; actually, perhaps better, since we tend to have a slightly higher media and page size than CorporateWatch – as shown in figure 2 where we come

second in the list.

Perhaps the best way to illustrate the placement of different sites in the final results is to take a step backwards in producing the BI score. Let's just look at the contribution of the three ratio figures to the final result (note the character ratio is inverted so that it works in the same sense as the other two ratios, and all work opposite to the final BI score – i.e., low is good, high is bad):
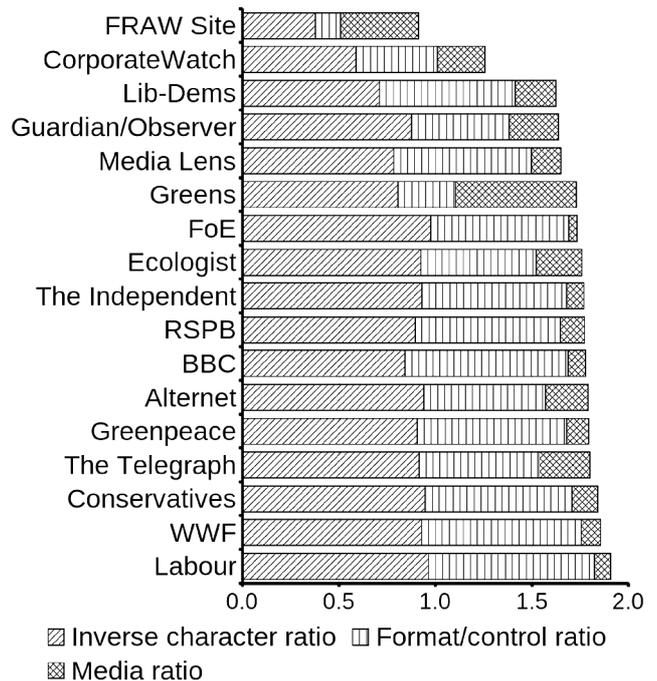

Figure 6. The three ratios and the bloat score

The effect of the high weighting on the character ratio is that it dominates the overall result – very efficient HTML mark-up, with a high level of visible data, gives a good result. For average web sites, designed to ape the latest web design standards down to the smallest dot and comma, it means that they are pri-

marily differentiated by the level of the formatting and control code within the pages – and it's the most poorly designed sites, with the largest files, that tend to have the most bloated code (as demonstrated in figure 3).

This idea is probably best illustrated with another histogram which averages the 'functional' elements of a page design across the 170 pages – as shown in figure 7.

The top graph shows the relative proportions of the page data, media and formatting data. Note that the left side is predominantly data and media; most of the formatting and control data is over on the right. That's because the trend for most pages is that more than half of their content is formatting and control data, not "visible" data.

Figure 7. Histogram of page components

We can then drill down another level by pulling apart the formatting and control figure to look at the contribution of the style sheet (*.css*) and script (*.js*) files to the overall value. Here again we see a similar divergence, indicating that much of the data is scripting, and the minority is style sheet data. There's a very simple reason for this – *the widespread use of web content management systems*[34] (such as Dreamweaver, Drupal, WebDev, Joomla, or Wordpress for blogs). Whilst humans might put a lot of effort into developing excellent content, the formatting of the content by a machine intelligence doesn't necessarily involve the same rigour. Many of these systems require little understanding to operate, and so in evolving the content for the pages the contradictions between the form of the content, and the efficiency of design, are never resolved. Most importantly, content management systems often include multiple or redundant formatting and code, because they don't have the capability – unlike human designers – to interpret the complexity of design and thus reduce the repetition and obsolescence inherent in computer-generated web pages. This lack of understanding is also reflected in the debate on design. Browsing around web design forums, whenever web design is discussed it's always related to the ergonomics of human interaction[35], never the energy and resource implications of the design strategy.
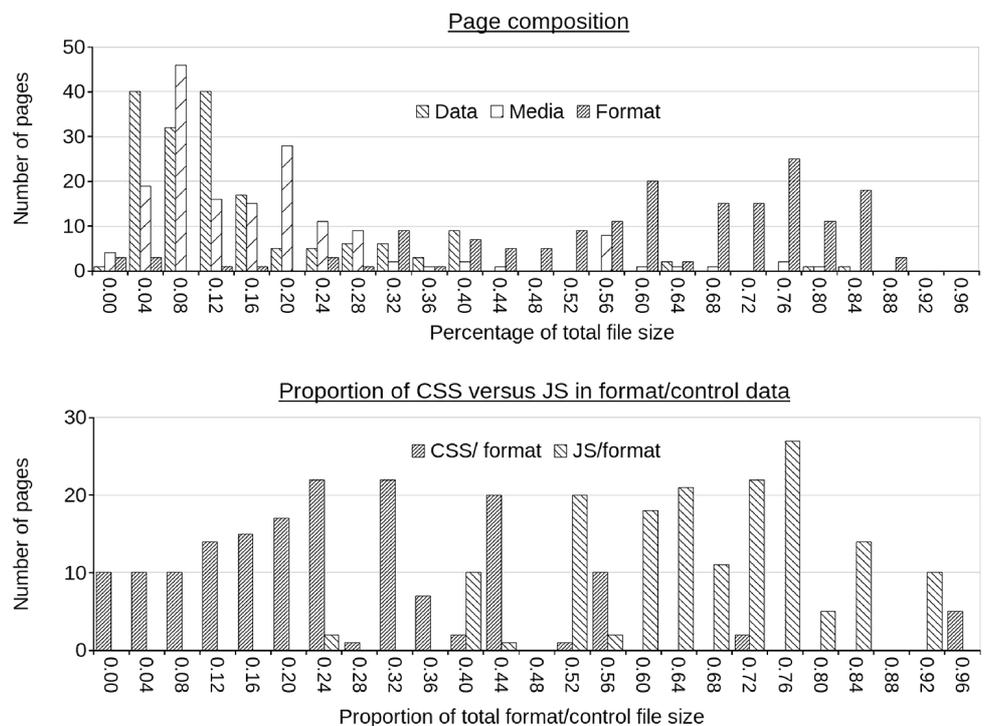
In many ways, the web, and web design, today is like the sweet rack at the super-store checkout – *it encourages you to take something "sweet" that you don't necessarily want just because it's convenient to do so.* Bringing in my other principal work interest – the social and economic impacts of energy and resource depletion – I think that whole agenda will come under great pressure over the next few years. We're already seeing that the growth in social network users, such as *twitter*[36], is reaching a plateau[37] – perhaps indicating a more general problem as people struggle to find the time to live in the real and the on-line world simultaneously. As the difficult economic times, and especially the rising energy costs of operating bloated data centres[38], begin to bite I think we'll see a greater pressure for more efficient design.

Whilst the improving efficiency of server hardware reduces the energy impacts of Internet use per unit of data, the growth in networks is still outpacing these savings – thus hdriving the energy demands of the virtual world[39] even higher. For example, in relation to the mobile phone networks study earlier, the improved UMTS[40] allowed greater traffic from the networks, but in turn, in part due to the lower costs of greater system efficiency, traffic has grown requiring that a much larger network be created (a classic case of the rebound effect[41]). If we want a more sustainable Internet the present narrow focus on energy consumption instead of the life-cycle impact of the

hardware is a problem[42]. Controlling, to some extent, energy demand is just one factor in costing new server purchases. One of the cost-efficiencies of installing new, lower-power servers isn't to reduce energy demand – it's to increase the capacity of the data centre and thus maintain the level of service or expand traffic. The whole "green IT"[43] idea hasn't been a matter of choice or commitment, it's a necessity to handle the ever greater power demand, and the need to dissipate all the heat that it produces[44].

If we could radically control the demand on servers, and on the power that the servers and network links consume, without changing the hardware, then that would represent a greater level of productivity. Of course, within present economic realities, that's only viable if network traffic ceases its inexorable growth – but that might be the reality we see emerging from current trends in Internet use, and also the trends within the global economy. If so, then the past myopia on the issue of IT productivity will come to a necessary end; at that point the management of the historic bloat of both software and data will become the "low hanging fruit" of efficiency – from which we can gain extra data capacity without having to change the nature of the hardware that supports the system.

Of course, in our own little virtual domain within cyberspace, we don't have to wait to make these changes. The little details make all the difference. Whilst I haven't gone into detail, the same bloat problem exists within HTML-formatted emails[45] – why use them when plain text provides an adequate option for most uses? Likewise, if you avoid proprietary software and its demands for constant upgrades, and use free software instead[46], you can keep your existing hardware running for longer, and use the same programs for longer. Also, due to the diversity inherent within the Linux operating system, it's possible to get low-bloat software to meet the restricted capacity of older machines[47].

What this really comes down to is understanding the technology that we use. I don't mean "read the manual", as often they tell you little in any case. I mean developing a deeper understanding of what technology, in its broadest sense, is, and what it does within our lives. As the peak of oil[48] and other essential commodities presses our decision-making, the first step to negotiating your way through the present and future difficulties in our relationship with technology is to understanding what it does, and how it achieves that end; and, once you have that understanding, you're free to consider other options to solve that problem. You may even decide that you no longer need a certain technology/the activity that it supports any more as, with the greater understanding you now possess, you might might decide that you want to seek a completely different solution altogether[49].

I'm just about finished now. I have the answers I need, some interesting graphs to illustrate the key trends, and some new ideas of how this "simplicity" approach within web design might be able to develop further. As I think of this I think about that great proponent of simplicity, Gandhi; and I ponder Gandhi's warning about unchecked consumption[50] –

> God forbid that India should ever take to industrialization after the manner of the West .The economic imperialism of a single tiny island kingdom (England) is today keeping the world in chains. If an entire nation of 300 million took to similar economic exploitation, it would strip the world bare like locusts.

This is where we are today. It's in part the rise of consumption in India and China, meaning that a far greater proportion of the world's population are finally – after the inequitable domination of the West – putting a greater demand on resources, which is creating the current pressures on everything from rare earth metals to food. The fact that, generally, we're also reaching the Earth's ecological limits is the most problematic part of our current commodity supply/price problems.

Set against the techno-utopianism[51] of the web and all it's dynamic services, the observations made here may seem a little awry – perhaps irrelevant to the way networked services are designed and constructed today. In fact, such arguments against the current "way of things" in the on-line world could be dismissed as a form of *digital Luddism*[52] (it is, after all, the 200th anniversary of the Luddites[53]). The use of the word "Luddite" is an easy method to dismiss all this, but this analysis goes to the heart of the efficiency and productivity of the Internet, and the on-line systems that modern society increasingly relies upon. Rather like the debate over transport being reduced to an issue of which fuel is put in the tank[54] rather than how we travel, today the debate on Green IT has become a discussion about power consumption, not the efficiency of the way we use the technology. Yes, the efficiency of services has a relationship to energy consumption; more generally it has a bearing on the way these services are designed and used – from the scale of data storage, to server capacity, to the utilisation of bandwidth and the upgrading of both hardware and software.

FRAW has been designed to be as efficient as we can make it, whilst serving the purposes for which it exists. That, hopefully, is demonstrated by the analysis provided earlier. We hope that others are also able to take these ideas and develop their own analysis of the web efficiency issue, and begin to address this much neglected problem; *measures such as productivity are at the root of ecological ICTs!*

References

1. *Command-line interface* – `http://en.wikipedia.org/wiki/Command-line_interface`

2. *FRAW Design Statement* – `http://www.fraw.org.uk/fraw/fraw_design_statement.shtml`

3. *Beyond the IT Productivity Paradox*, Prof. Leslie Willcocks and Stephanie Lester, European Management Journal, vol.14 no.3 pp.279-290, June 1996 – `http://dx.doi.org/10.1016/0263-2373(96)00007-2`

4. *Software bloat* – `http://en.wikipedia.org/wiki/Software_bloat`

5. *Media owners urged to double file sizes for on-line ads*, mUm-BRELLA, 2nd February 2011 – `http://mumbrella.com.au/media-owners-urged-to-double-file-sizes-for-ads-38906` ; see also *Aussie advertisers call for more bloat in web ads – They're coming for your bandwidth*, Richard Chirgwin, The Register, 2nd February 2011 – `http://www.theregister.co.uk/2011/02/02/increase_ad_sizes/`

6. *More Firefox Bloat? Say It Ain't So, Mozilla*, Scott Gilbertson and Michael Calore, Wired, 17th May 2007 – `http://www.wired.com/software/coolapps/news/2007/05/firefox_bloat`; *update 3/6/11*, see also *Google to abandon older browsers*, BBC News On-line, 3rd June 2011 – `http://www.bbc.co.uk/news/technology-13639875`

7. *Criticism of Top Gear* – `http://en.wikipedia.org/wiki/Criticism_of_Top_Gear`

8. *Total Quality Management: Issues and methods for measuring productivity for information systems*, Richard McCarthy; pp.465-478 of *Systems Development Methods for Databases, Enterprise Modeling, and Workflow Management*, Wita Wojtkowski, W. Gregory Wojtkowski, Stanislaw Wrycza and Joze Zupancic (eds.), Springer, February 2000. ISBN 9780-3064-6299-3. £170.

9. *ZX80* – `http://en.wikipedia.org/wiki/ZX80`

10. *Modem* – `http://en.wikipedia.org/wiki/Modem`

11. *The Story of Electronics* – `http://storyofstuff.org/electronics/`

12. Urban Dictionary: *Kipple* – `http://www.urbandictionary.com/define.php?term=kipple`

13. *Moore's Law* – `http://en.wikipedia.org/wiki/Moore%27s_law`

14. *Life Cycle Assessment of the Mobile Communication System UMTS – Towards Eco-efficient Systems*, Mireille Faist Emmenegger, Rolf Frischknecht, Markus Stutz, Michael Guggisberg, Res Witschi and Tim Otto, The International Journal of Life Cycle Assessment, vol.11 no.4 pp.265-276, 2005 – `http://www.fraw.org.uk/f.html?faist2005`

15. *Global Information Society Watch 2010*, Association for Progressive Communications/Hivos, December 2010; specifically see the introduction, and the UK country report, which I wrote for GISW 2010 – `http://www.fraw.org.uk/mei/papers/index.shtml#gisw`

16. *Electronic waste* – `http://en.wikipedia.org/wiki/Electronic_waste`

17. *Consumer culture keeps carbon emissions high – Consumer goods are not as cheap as their price tags suggest, because they lead to significant emissions elsewhere*, Janet Pelley, Journal of Environmental Science and Technology, vol.43 no.16 p.6116, 24th June 2009 – `http://pubs.acs.org/doi/pdf/10.1021/es901774u`

18. *Smarter bytes, slimmer footprints*, Bill Thompson and Jon Wallace, Green Futures, 13th October 2008 – `http://www.forumforthefuture.org.uk/greenfutures/articles/Smarter_bytes_slimmer_footprints%2B`

19. *Gartner Estimates ICT Industry Accounts for 2 Percent of Global CO2 Emissions*, Christy Pettey, Gartner (press release), 26th April 2007 – `http://www.gartner.com/it/page.jsp?id=503867`

20. *Impacts of Information and Communication Technologies on Energy Efficiency: Final Report*, Adrian Breton, Bio-Intelligence Service for the European Commission, 2008 – `ftp://ftp.cordis.europa.eu/pub/fp7/ict/docs/sustainable-growth/ict4ee-final-report_en.pdf`

21. I've listed all the details of the pages selected, the measurements and analysis of the data in the spreadsheet on which this work is based; available from `http://www.fraw.org.uk/mei/ecolonomics/01/ecolonomics-012-statistical_data.pdf`

22. *Firefox* – `http://en.wikipedia.org/wiki/Firefox`

23. *HTML element* – `http://en.wikipedia.org/wiki/HTML_element`

24. *Static web page* – `http://en.wikipedia.org/wiki/Static_web_page`

25. *Web template system* – `http://en.wikipedia.org/wiki/Web_template_system`

26. *Dynamic web page* – `http://en.wikipedia.org/wiki/Dynamic_web_page`

27. *Cascading style sheets* – `http://en.wikipedia.org/wiki/Cascading_Style_Sheets`

28. *Client-side scripting* – `http://en.wikipedia.org/wiki/Client-side_scripting`

29. *JavaScript* – `http://en.wikipedia.org/wiki/Javascript`

30. *Server-side includes* – `http://en.wikipedia.org/wiki/Server_Side_Includes`

31. *In-line frame* – `http://en.wikipedia.org/wiki/Iframe`

32. *Web analytics* – `http://en.wikipedia.org/wiki/Web_analytics`

33. *Histogram* – `http://en.wikipedia.org/wiki/Histogram`

34. *Web development* – `http://en.wikipedia.org/wiki/Web_Management`

35. *User interface design* – `http://en.wikipedia.org/wiki/User_interface_design`

36. *Has Twitter reached its peak? – Micro-blogging service Twitter's user growth has almost levelled off since September 2009, according to a study*, Mercedes Bunz, Guardian On-line, Friday 12th March 2010 – `http://www.guardian.co.uk/media/pda/2010/mar/12/twitter-growth`

37. *Social Networking Growth Set To Peak*, Google Live Search, 18th March 2011 – `http://www.googlelivesearch.com/web-news/social-networking-growth-set-to-peak/`

38. *Data centre energy costs a rising concern – Companies need to address their data centre energy use so it does not become an escalating problem, Gartner says*, Tom Brewster, ITPro, 29th Sep 2010 – `http://www.itpro.co.uk/627289/data-centre-energy-costs-a-rising-concern`

39. *The Internet: One big power suck*, Steve Hargreaves, CNN-MoneyTech, 9th May 2011 – `http://money.cnn.com/2011/05/03/technology/internet_electricity/`

40. *UMTS* – `http://en.wikipedia.org/wiki/Universal_Mobile_Telecommunications_System`

41. *Rebound effect* – `http://en.wikipedia.org/wiki/Rebound_effect_%28conservation%29`

42. *The Green IT movement has lost the plot – E-waste, and not energy conservation, is what really matters*, David Moschella, CompterWorld, 18th May 2011 – `http://www.computerworld.com/s/article/9216828/The_Green_IT_movement_has_lost_the_plot`

43. *Green computing* – `http://en.wikipedia.org/wiki/Green_computing`

44. *Greening the data centre not just about the environment*, Pieter le Roux, IT News Africa, Thursday 26th May 2011 – http://www.itnewsafrica.com/2011/05/greening-the-data-centre-not-just-about-the-environment/

45. *HTML email* – http://en.wikipedia.org/wiki/HTML_email

46. *Free and open source software* – http://en.wikipedia.org/wiki/Free_and_open_source_software

47. *Lightweight Linux distribution* – http://en.wikipedia.org/wiki/Lightweight_Linux_distribution

48. *Peak oil* – http://en.wikipedia.org/wiki/Peak_oil

49. *Voluntary simplicity* – http://en.wikipedia.org/wiki/Voluntary_simplicity

50. *Why Gandhi Still Matters*, Gandhian Institution (undated) – http://www.mkgandhi-sarvodaya.org/articles/gstillmatters.htm

51. *Technological utopianism* – http://en.wikipedia.org/wiki/Technological_utopianism

52. *Luddite* – http://en.wikipedia.org/wiki/Luddite

53. *Luddite bicentenary* blog – http://ludditebicentenary.blogspot.com/

54. *Food vs. Fuel* – http://en.wikipedia.org/wiki/Food_vs._fuel