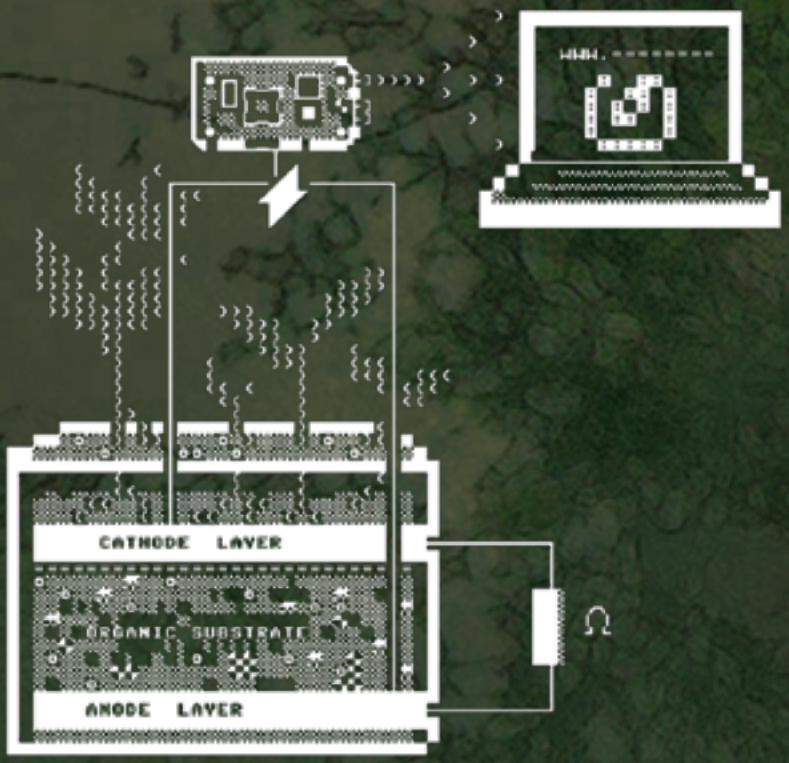
compost computer

More-Than Human Design through Critical Climate Computing

The Low Carbon Website Toolkit

This toolkit will guide you through the steps of how to create a Microbial Fuel Cell (MFC) -powered webserver. This is a low-cost, low-power solution for hosting websites using compost as a power source.

Who are we?
Critical Climate Computing is a research group based at the University of the Arts London. We work to advance research and creative practice that explores topics such as low-carbon and de-growth computation, permacomputing, and green-transition technologies within creative and critical arts and design practice.



Why have we made this?
This toolkit is part of a UK Arts and Humanities Research Council-funded project called More-Than-Human Design Through Critical Climate Computing. This project's aim was to produce a proof-of-concept compost-powered webserver, and redesign the UK arts and culture organisation FutureEverything's website using low carbon design logics. We have been working in collaboration with the Manchester-based urban growing cooperatives Sow the City (StC) and Manchester Urban Diggers (MUD), and FutureEverything (FE). Through these collaborations we developed an MFC-powered webserver fuelled by MUD's compost, and used it to host a new low-carbon website that we designed for Future Everything.

compost computer

More-Than Human Design through Critical Climate Computing

The Low Carbon Website Toolkit

Glossary of terms

Server/Webserver

Software that runs on a computer that distributes web content via the HTTP protocol. It 'serves' the content of a website (stored as files on the computer) to the wider internet, where it can be accessed through browsers that connect to it online.

Client

In website terminology, the term server is used to describe the owner of the website, and 'Client' is used to describe any machine viewing the website from outside via a web browser.

Compute

In web design, 'compute' refers to the processing power and operations required by both the server and the client device to render, display, and interact with a website. This includes tasks like executing JavaScript, processing server-side code, rendering graphics, and handling database queries.

Carbon cost

The carbon cost in web design refers to the total greenhouse gas emissions generated throughout a website's lifecycle, encompassing the energy consumed by data centers (servers), network infrastructure for data transfer, and end-user devices for processing and viewing the content. It essentially quantifies the environmental impact of a website's existence and usage.

Introduction

Website energy consumption centers around three key elements:

1.The Server

Every time a user requests a webpage, the server has to retrieve and send all the assets (HTML, CSS, JavaScript, images, videos, fonts). The more data involved, the more work the server's CPU, memory, and network interfaces have to do, and the more energy is consumed.

2.The Network

When a website is accessed, the data on the server travels through countless routers, switches, fiber optic cables, and other network equipment between the server and the client. Each piece of equipment consumes electricity. The more data that needs to be moved, the more sustained energy consumption occurs across the network infrastructure.

3.The Client

When accessing a website via a device such as a laptop, phone, etc., the client expends

The Low Carbon Website Toolkit

energy to download the website's data via Wi-Fi or mobile networks. Larger files mean longer download times and more energy consumption by the network interface. Once downloaded, the client device's CPU and GPU consume energy to process JavaScript, render the HTML and CSS, decode images and videos, and display the page on the screen. The more complex the page (more JavaScript, higher resolution media, intricate animations), the more processing power (and thus energy) is required.

In essence
More Data = More Energy for Transfer: Whether it's from the server to the client, or across the network, larger amounts of data require more electrical energy to transmit.

More Data (and complexity) = More Energy for Processing: Both servers and client devices need more energy to handle, decode, execute, and display larger and more complex datasets.

A key strategy for reducing carbon load is therefore to reduce the amount of data passed between server and client, thus reducing the carbon cost of transferring this data between devices.

While there are a number of server-side changes that can be made for reducing the carbon cost of a website (such as server-side caching to reduce the number of times an asset is loaded), this guide will focus on those practices that can be affected specifically through coding and web design practices.

Code Languages
Modern websites often rely on Javascript for features such as animations or in-page applications. Sites designed with Javascript require computational resources to be spent on both the server side and the client side, adding to download size and subsequent network energy for transfer, leading to higher overall energy consumption across user devices, networks, and servers. The computation required to run Javascript also means that both low power servers and older or low-powered browsers often cannot serve or load Javascript. While Javascript has become common across the web, it is not essential, and it's one of the main reasons of web bloat due to third-party scripts and extremely compute-heavy cosmetic and interactive 'add-ons'.

To navigate this, we recommend designing a website using only HTML and CSS. These are huge, well-established libraries that offer a lot of freedom and control. Their longevity as the backbone of the internet makes them extremely reliable, longlasting and backwards compatible, without the additional compute cost of Javascript operations.
An example of this is the content-aware image editing tool developed by Mariana Marangoni for our 'More-Than-Human Design Through Critical Climate Computing' project.

compost computer

More-Than Human Design through Critical Climate Computing

The Low Carbon Website Toolkit

Lazy loading
Lazy loading is a web optimisation technique where non-critical resources (like images, videos, or iframes) are not loaded until they are actually needed, typically when they enter the user's viewport.

In HTML/CSS web design, this can be utilized by adding an loading="lazy" attribute directly to <img> or <iframe> tags in the HTML. This significantly reduces initial page load times and bandwidth, as the browser only fetches content that the user is likely to see immediately.

Fonts
When a website uses a custom font, the client first downloads the font file(s) from the server to the client's device. Once downloaded, the browser then renders the text using that specific font. This adds an additional file download to view the webpage, increasing the amout of data that the page uses. To reduce carbon cost, prioritise using system fonts (fonts already installed on the user's device) where design allows, as they require zero download. Common system fonts include

Serif Fonts
- Times New Roman
- Georgia

Sans-Serif Fonts
- Arial
- Helvetica (very common on macOS/iOS)
- Verdana
- Tahoma
- Roboto (often default on Android, widely supported by Google)
- Open Sans (similar to Roboto in wide adoption)
- Segoe UI (Windows default)
- SF Pro Display/SF Pro Text (Apple's system font for macOS/iOS)
- Cantarell (GNOME desktop environment)

Monospace Fonts
Courier New
- Consolas
- Menlo
- Monaco

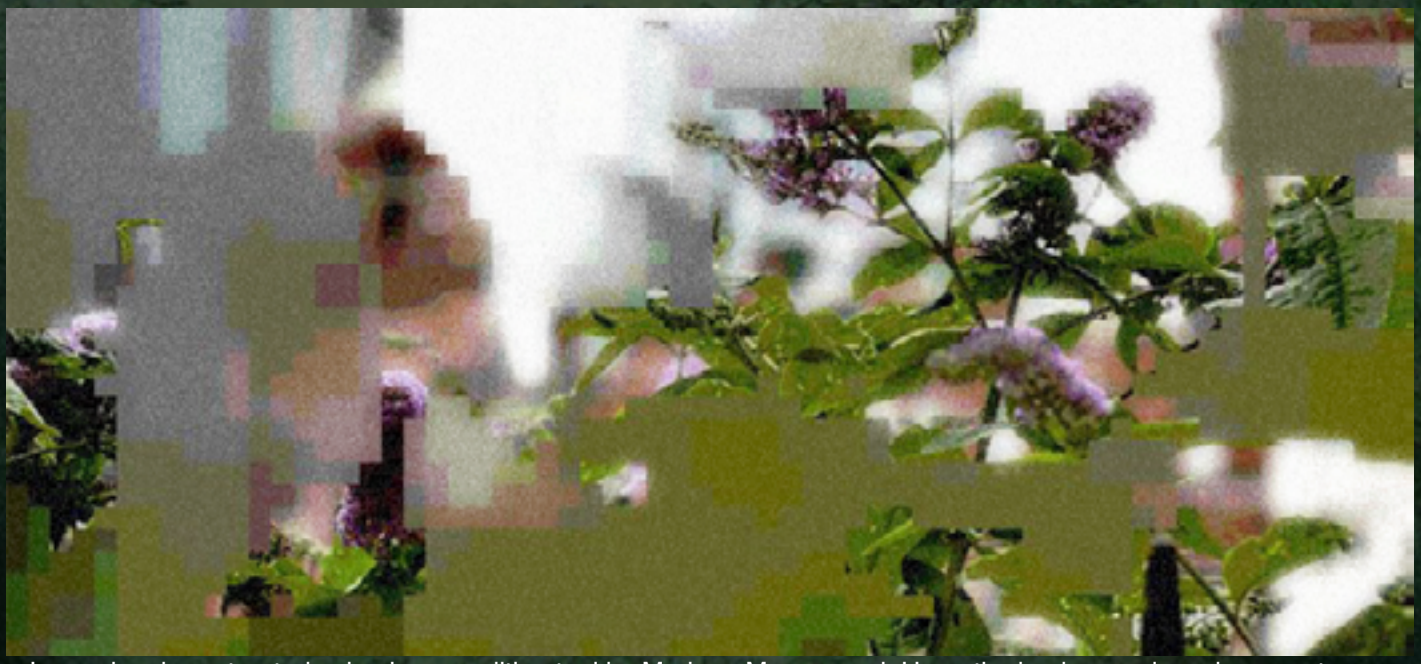
compost computer

More-Than Human Design through Critical Climate Computing

The Low Carbon Website Toolkit

Media

Image hosting and image loading can consume a substantial amount of power in a website, at both the host and client side. The bigger the filesize the image, the more power needed to both serve the image and download it on the client side. Modern websites can often be image-heavy, and while image formats such as .avif and .webp can be used to compress file size down as much as possible, there are other approaches that can be used to help reduce filesize further.



< Image has been treated using image editing tool by Mariana Maraongoni. Here, the background can be seen to have a more 'pixelated' effect, while the central detail of the plant is largely untouched. This process reduces visual complexity and filesize, while retaining core features and readability>

This custom software tool significantly reduces the image's file sizes while giving them a unique visual signature that stands out from other widely available tools for pixelating and dithering images available online. The technical process behind the tool is that it image functions by duplicating an uploaded image to a new canvas where each pixel in any given cluster of cells is replaced with the average of the surrounding pixels, massively reducing visual detail and pixel variance. A limited number of representative colors is extracted using a simple K-means clustering algorithm, and each pixel's average is mapped to the closest color in this reduced palette.

compost computer

More-Than Human Design through Critical Climate Computing

The Low Carbon Website Toolkit

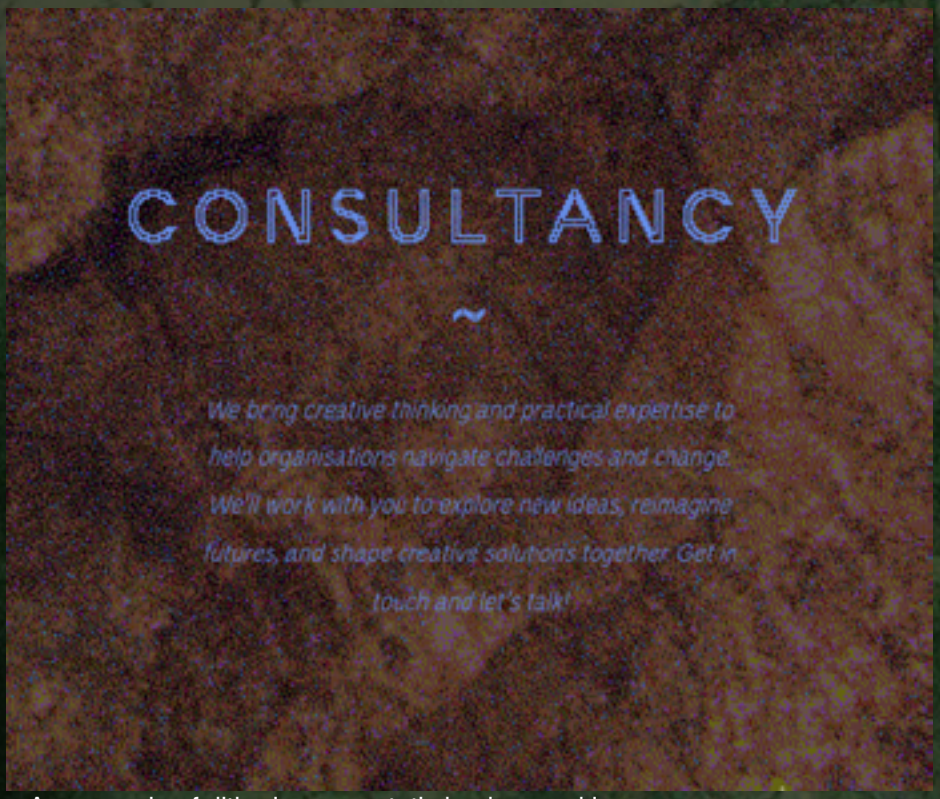
In simple terms, this process compresses or ‘flattens’ the image’s background features such as a background sky or wall while most detailed areas of the image, such as objects or figures in an image’s foreground, are preserved. Another example can be seen in this image below, taken from the FE website that runs on the Compost Computer server

< Image has been treated using image editing tool by Mariana Maraongoni. Here, the foreground and the background can be seen to have a more ‘pixelated’ effect, while the central detail of the image remains legible. This process reduces visual complexity and filesize, while retaining core features and readability>

In our own experiments in low carbon web design, we also explored practices such as dithering to reduce the filesize of background images and textures on a website. Dithering creates a low-res version of an image by strategically placing pixels of different available colors next to each other, creating an illusion of depth and variety.

The Low Carbon Website Toolkit

This technique can significantly reduce the carbon cost by decreasing the image file size (as fewer colors require less data), thus reducing the energy needed for both transferring the image over networks and the client device's processing and rendering of the smaller, simpler file.



< An example of dithering on a static background image>

Content Delivery Networks and Media-Heavy Websites

Approaches such as these can be explored to reduce filesize wherever possible in images hosted on low-carbon websites. Of course, reducing the number of images, videos, or other files will further reduce the computational cost of serving and loading a webpage, but it is not always practicaly to do this depending on what type of website you're building and for what purpose. If you're running a website on a very low-power, low-storage system (such as the Seeed XIAO nRF52840 series server device that we used for the Compost Computer project, which has a total storage of 2MB), then using a Content Delivery Network (CDN) to host images and videos may be a solution. This can be as simple as a static website, hosted on a traditional server, which hosts all the images and videos which are then loaded from that source rather than saved and loaded from the low-storage server.

However, if your goal is to reduce both server load and client load, then using a CDN still means that the client will download the images and media for the site, just not from your

compost computer

More-Than Human Design through Critical Climate Computing

The Low Carbon Website Toolkit

own hardware. This would likely not meaningfully reduce the overall carbon impact of the website, especially if your CDN is hosting high-resolution images and video that are embedded on your site. There is no quick and easy solution to this, and striking the balance between media content and carbon footprint should be assessed on a case by case basis, on reflection of the goals of the website, the restrictions of the hardware, and the desire to reduce the carbon impact of the site.

If you want your site to use images and video, but are willing to explore techniques for reducing filesize of your media, there are some options to explore. One is Handbrake, a free, open-source utility that allows you to compress a video file into an incredibly small video file. Avidemux is another open-source video editing and compression tool that is free to use. It can be used to compress video files to under 1MB a minute.

Resources

Guides to good practice in low-carbon web design

https://jeffhuang.com/designed_to_last/
https://solar.lowtechmagazine.com/low-tech-solutions/
https://sustainablewebdesign.org/estimating-digital-emissions/
https://michaelandersen93.substack.com/p/greening-the-web-a-study-on-low-carbon
https://climateaction.tech/actions/create-low-carbon-images/

Carbon measure/web sustainability resources:
https://www.websitecarbon.com/
https://ecograder.com/
https://sustainablewebdesign.org/
https://www.wholegraindigital.com/blog/sustainable-web-design-book-updated/

Credits

Project Lead and Management – Wesley Goatley and Eva Verhoeven
Project concept - Mariana Marangoni, Lucy Sollitt, and Shinji Toya
Project ideation and research –Mariana Marangoni and Shinji Toya
Web design – Mariana Marangoni and Alistair McCloymont
Back-end server architecture lead – Mariana Marangoni
MFC power system and physical computing lead – Shinji Toya

Technical support and consultation:
Felix Loftus, Karsten Goodwin – Specialist Technicians, Central Saint Martins College of Art and Design, Physical Computing Lab
Agnes Cameron - Specialist Technician, Creative Computing Institute
Alistair McClymont and Kevin Lee - additional web development

ual: university of the arts london

Future Everything