

somethingdark

HTML vs. Flash

considerations when creating a
web magazine

Our single-minded objective was to create, from the ground up, a new generation of web-based publication that combines the dynamism of the internet with a design and reading experience that replicates as closely as possible that of a print publication. We had to be innovative in our approach, both technologically and with our content, to ensure our project's relevance to contemporary culture.

There were a number of problems to solve in achieving our objective, the most basic being which medium would be best suited to deliver this *webmagazine* – specifically, whether to use a flash player, or whether to apply ourselves to HTML-based coding. After much research and consideration, we chose HTML. Let us provide you with an insight as to why.

Budget

The major issue that affected our planning process was actually budget – specifically, what resources could we afford to buy in that would work as we needed, and in how we promote *SomethingDark* webMagazine once launched. With this, we investigated the established trend in online publishing, that is, “flash” magazine software, and soon realised the most dynamic and very expensive high-end versions as much as the ubiquitous generic versions would not meet our needs for the way we wanted to include additional information outside the magazine proper, such as actively linked article footnotes and resources at the bottom of each page.

Why spend a large amount of money on a solution that's not going to be able to achieve what you require and that isn't scalable, when you can create your own, more powerful solution?

Optimisation

Flash-driven online publications are almost invisible to search engines regarding optimisation; rumours that search engines can read and index flash are irrelevant – they still can't index in their own right the individual pages of the magazine that are locked into flash files. Thus, while flash can look impressive, we're convinced that, by using HTML to it's maximum potential when it comes to web publishing, we have created a true web-based magazine that is not only more dynamic in its functioning and internal navigation, but is more sustainable with respect to vital search-engine indexing.

For example, a page on a particular subject in *SomethingDark* webMagazine will also be its own web-page, meaning that it can be indexed as an individual entry on the search engines. Links from the search engines on a particular subject can link straight to that magazine page: no waiting for the flash magazine to load, and no having to flick through the pages in the flash player looking for the relevant content. That is, with HTML, the relevant content is on screen immediately – superb for general ease-of-use, whether the application is for serious research or for casual browsing.

Our promotional budget was word-of-mouth; however, talk is cheap. Considering this, the magazine, upon launch, needed to promote itself – and mainly via the search engines. With flash, indexing would have been minimal – a few entries – whereas HTML gives the search giants a great deal of accessible content: *SomethingDark* (SDK) site statistics continuously confirm that we have achieved a number of individual page listings on Google that flash-based sites could never hope to achieve.

Accessibility and portability

Flash works very well – a website can look like a print publication on screen, even mimicking the turning of pages and allowing zoom-in – but HTML is much more accessible, considerably reduces demands on bandwidth, and, in our case, allowed us to add a great deal of dynamic content.

As those who rely on such accessibility for enjoyment of their everyday use of the internet know – through experience if not technical knowledge – screen-readers and related software struggle to work cohesively with flash: this is because they are built for HTML. Additionally, all modern web-browsers now have to comply with accessibility guidelines for HTML-coded sites; this means functions such as navigation and zooming are allowed through the use of access-keys, for example. Thus, with a few key-strokes, a correctly programmed HTML site will operate just as you would expect a flash site to operate.

As an HTML-coded magazine and website, SDK has shortcut keys: screen-readers can skip through it and read the content. And, furthermore, *SomethingDark* webMagazine can be viewed on the [Apple iPad](#) – unlike flash magazines, which require the purchase of special plug-ins. We are also into our second generation of the webmagazine for web-capable smartphones.

Dynamic data

SomethingDark webMagazine is also much more than a print magazine that's been given a website or rendered into a flash version for the web, and this is where the fact we're using HTML and are determined to make maximum use of being a *webmagazine* really comes into its own. We are, for example, building an expanding range of associated directories alongside the actual webmagazine, the two principal ones being our Contributor Directory and our Resource Directory.

The [Contributor Directory](#) allows every contributor to the magazine to build a detailed profile, including a biography, portrait photo, various sections for listing a wide range of professional and personal achievements, a gallery of images for our visual artists, links to their own websites, and an index of all their contributions to *SomethingDark* webMagazine with direct links to individual items of content irrespective of the issue or page of the magazine.

The [Resource Directory](#) is an innovative feature of SDK that allows us to associate selected written content in our magazine with relevant, quality resources that are either referred to directly in *SomethingDark* webMagazine or contextually important to the subject matter. Each resource in this directory has a description, a link back to its own main website, and, again, dynamic links to every page in every issue of the magazine relevant to that resource.

The utility of our directories does not end with the functioning mentioned above: all the data associated with a particular directory entry is stored in a database and can be extracted wherever relevant and displayed on-screen – another benefit of creating an HTML-based website. And where else to place this wealth of information than below the relevant page you are looking at whenever you are viewing the webmagazine – open a page, scroll down and you'll see footnotes relating to the content on that page, and that page only; and a list of all contributors and resources in the magazine issue being viewed, as well as the ones relevant to the current page being highlighted.

These systems of internal and external linking are, as indicated above, extremely effective for search engine optimisation (SEO). It also means that as SDK grows, both contributors and visitors will increasingly see the benefit of our HTML-based magazine.

Efficiency

We've been efficient in coding the site and, as you now know, very efficient in how we store and link that information to extract and utilise the data to its fullest extent, both for us and for SDK's readers. We are well aware of how much "information" is available on the internet: wading through swamps of information can be frustrating and can actually hinder productivity rather than aid it, and that is why one of our key objectives was to provide carefully selected, complementary information and relevant resources that are linked to the core original content of our magazine in ways that maximise accessibility. In this way, the resource value of SDK is multiplied.

Considering the above, all of this additional information presented on screen and regularly retrieved from a database certainly makes for a resource-hungry website. Could flash have won

this battle? The answer, as simple as it is direct, is a resounding “No”. Flash is a relatively “flat” platform with limited dynamism, and thus cannot work with and utilise the information it’s fed as well as an HTML-based alternative. The result is that flash-based sites, at best, might deliver no- (or little)-additional-data relatively quickly – but this is not better than delivering substantial-additional-data not quite as quickly. On the other hand, SomethingDark delivers great-additional-data, and very fast. How? Let us introduce our “caching engine”.

Our caching engine is a special bit of kit; although not a new concept since large corporate websites have been using them for years, they’re relatively new in the realm of online publications. This is our hidden weapon we hold against flash.

The reality of a highly database-driven website is that potentially significant server resources are used every time you retrieve information from the database (not even including the actual page content such as images): thousands of records and bits of information are processed every time a website page is loaded, and this can add up to a good few seconds. Not good for our modern website visitors, who, rightly, expect a page and expect it *now*.

What our caching engine does is load our fully-functional page – the dynamic, database-driven resources and contributor information included. It does take a few seconds, but once it’s done all of this work once, and knows exactly what resources are needed for a particular page, it saves the required data – and *only* the required data – to a cache file. The result of this is that when a visitor to the website requests an information-rich page, the website simply accesses the cache file and dumps it to screen: the database isn’t touched, but all the relevant data is presented to the viewer in a few hundredths of a second.

We think we’re at the forefront of a dawning era for web-based publications – we’re even an [iPad-compatible magazine](#) – and that *SomethingDark* webMagazine does indeed represent “the new wave”.