



European Public Sector Information Platform

Topic Report No. 2015/01

Interface Design Strategies for Data Portals

Author: Ana Isabel Carvalho, Ricardo Lafuente

Published: January 2015

Table of Contents

<u>Keywords:</u>	3
<u>Abstract/ Executive Summary:</u>	3
2 Introduction	5
3 The role of interface	6
3.1 Complexity	6
Get away from the data structure.....	6
Technical terminology	7
Don't include sections for the sake of it.....	8
Simplify aggressively	8
What you don't show is as important as what you show.....	9
3.2 Layout	9
Visual hierarchy	9
Question all defaults	10
Wasteful elements	11
Embrace the grid	11
Be careful with re-designs.....	12
3.3 Typography	12
Type plays a role.....	12
Typeface choice.....	13
Typeface combinations	13
Web typographic standards	14
Get a designer's advice.....	14
3.4 User Interface	14
Reduce the number of steps from X to Y	14
Effective search	15
Platform independence.....	16
Help text.....	16
Every element is a promise	17
3.5 Web Technologies.....	18
Permalinks and longevity	18

	<i>Sysadmins</i> are crucial	18
	Social networks	19
	Proprietary technologies vs. standards	19
	Follow your metrics to see what the problem spots in your site	20
4	Conclusions and recommendations.....	21
5	References and resources.....	23
	About the Authors.....	24
	Copyright information	24

Keywords:

interface design, data portals, information design, interface patterns, usability

Abstract/ Executive Summary:

This report addresses use cases of official data portals: web sites with the main purpose of

providing access to public information, either in database form or other. This kind of sites usually faces the problem of engagement with its target audience (researchers and citizens), and it often happens that independent organisations step forward to provide alternatives to the official resources.

This report presents a set of strategies that can be considered and implemented to improve the clarity and usability of data portals, and it is aimed for anyone involved in the development, planning and implementation of such web sites.

2 Introduction

The target audience of this report comprises anyone who might be remotely related to the planning of a data portal web site, either if they are managing the project or involved in the technical implementation.

It is not common to feature a specialised design team in the endeavour of conceiving this kind of project. However, design concerns ought to feature highly on project leaders' priorities, since those will often **determine the success or failure in engaging citizens and researchers to interact with the new platform.**

This report addresses the use cases of official data portals: web sites with the main purpose of providing access to public information, either in database form or other. This kind of sites usually faces the problem of engagement with its target audience (researchers and citizens), and it often happens that independent organisations step forward to provide alternatives to the official resources.

During the course of this report, we will refer to these sites as “data portals”, even though many of the indications here may very well apply to a broader range of use cases, specifically more to general-purpose public information web sites. Care has been taken when formulating design principles in a way that is accessible and understandable to non-designers.

Data portals are particular kinds of sites in that they often deal with complex, dry and technical subject matter; offer access to an extensive array of datasets and resources, and usually require domain-specific knowledge of databases and the structure of government institutions.

The goal of this report is, therefore, to present a set of strategies that can be considered and implemented to improve the clarity and usability of data portals. We identify a set of tactics that can be employed to improve on and simplify the structure of a data portal. This report will be structured around five specific themes and areas of work which will then be divided into individual strategies; anyone in charge of a data portal can look through them and identify the points from which their site would benefit.

3 The role of interface

We believe that interface matters are central to the success or failure of this effort. By "interface", we mean "user interface" — in the field of human-computer interaction, this term refers to the means by which people manifest their intentions to machines, which then execute operations and/or return a result.

The idea of "interface" can be framed in a broader way: a government portal can serve as an interface between citizens and the central administration. However, **if not well planned, it might have the opposite effect**: by affirming the ability of closer contact but failing to deliver on the promise, it can sour the expectations of users with regards to the availability of government to establish contact.

Therefore, one should strive for **good and clean interfaces**. The main aspect of such interfaces is that they do not get in the way; ideally, they just won't even be noticed. This is parallel to the common saying that "good design is invisible". A well-designed bike makes us forget about it and just think of the journey. Likewise, a well-designed government portal can just become part of people's regular interactions; people stop thinking that they are interacting with the portal and simply interact with the institutions themselves. This thought shift is what one should aim for when planning a new portal.

3.1 Complexity

The essence of design is solving the problem of complexity. This can be done by hiding or making it simple through the use of a set of tactics.

Databases and datasets are inherently complex entities; therefore, it's vital that the issue of complexity is addressed when planning an interface to present them.

Get away from the data structure

When given a set of well-built database with dozens of different fields, it is tempting to go all the way and display all the information inside – after all, it makes sense to showcase the (usually unacknowledged) work of putting it together.

Sadly, too much information can be worse than no information at all. It is easy to get lost in a sea of data if users are presented with dozens of fields. And more often than not, getting lost is followed by a loss of interest.

Complexity is intimidating; no one enjoys the feeling of being overwhelmed and believing that they don't understand what they are looking at. This is a design issue: users have to be assured that the content of a data portal is readable and understandable.

What to do? It is not a trivial task to abstract away the complexity of data, and there certainly exist edge cases where that would be a hard task. Nevertheless, there are some general patterns to deal with this; the most straightforward one would be to avoid reflecting the internal data structure in the front-end. In other words: if tasked with displaying a database with twenty different fields of information for a single entity, it's best to start by considering the priority and importance of each field, reflecting it visually through emphasis. Less important fields can be pushed to the bottom of the page, have their emphasis reduced through visual priority cues, or even hidden — would the general public be interested in that information? This is often not the case with internal fields like “Last modified” or internal ID markers. Showing these will inevitably make a portal look more “database-like” and less human.

A practical example: on a page about a specific person, that person's name and picture should be well visible, their birth date can be pushed to a less prominent “Details” section, and the timestamp of the last time this person's info has been updated should be almost hidden in the bottom of the page, if at all present.

One of the simplest, yet most effective, design techniques would therefore be to **understand that the internal data structure and the public data presentation should not be equivalent.**

Technical terminology

One of the largest hurdles to inviting the interest of the general public in public information is the use of specialist terminology. This is made worse by the fact that the more one is embedded in a specific area of interest, the quicker one forgets that most people are not acquainted with the technical terms used.

When faced with things that they do not understand, people will inevitably be led to frustration. That's an effect that needs to be avoided, and the only way to get there is to **reformulate jargon into understandable terms.** And this is hardly an easy task — in most cases, the advice of a copywriter or editor should be sought in order to determine how to message existing copy into a form that can be catered to non-specialist audiences.

An easy test to conduct would be to present terms to laypeople (one doesn't need to look

beyond their own families!) and see if they can gather their meaning. If there isn't a quick answer, the term needs to be reformulated; even if the correct meaning is guessed or hinted at, that is not enough: **clarity is paramount**.

Don't include sections for the sake of it

The first **priority ought to be what the users of the site will be looking for**; only after you should consider the things that you want them to see.

One common occurrence is the prioritisation of a blog or news section over the dataset index. While the news section of a portal might be of relevance (e.g. for indicating new datasets), keep in mind that new users will probably be looking for the data itself, whereas the news will be relevant to frequent users; these frequent users will most probably bookmark the blog URL or subscribe to its RSS feed, doing away with the need to highlight this section on the front page.

We can apply this line of thought to other sections: **any technical resource should be delegated to a "developer" section**, lest the non-technical audience be alienated; primers, introductions and tutorials can be placed on an "About" or "Help" section. This will unclutter the main interface in order to emphasise the elements that should be prioritised.

Simplify aggressively

Minimalism is a hard design approach; this because simplification does not mean careless removal of elements, but rather the reduction to the essentials without becoming cryptic or incomplete.

When planning or reviewing the structure of a data portal, it is worth carrying out the exercise of asking why every single element is in the place it is. If a satisfactory reason cannot be easily pointed out, it is most likely a sign that its presence and graphic arrangement should be considered and maybe reformulated.

The benefit of an aggressive simplification approach is the removal of any unnecessary or low-priority elements. Uncluttering a page has the immediate benefit of making a site more user-friendly. A long page will always be less readable than a shorter one that doesn't require scrolling and manages to present all necessary information in one screen (of course, not all pages lend themselves to this kind of reduction.)

As we address the issue of long pages, with a significant amount of vertical scrolling, we take a

moment to discuss pagination. Pagination refers to the separation of content into different pages and connecting these pages with a "Previous/Next" button, or sequential numbering.

How to decide whether to present a large amount of information in a long page, or many different pages interconnected by pagination?

Pagination is adequate whenever the content can stand a separation; in other words, it is discouraged to divide content that should be shown as a whole. For instance, a transcript of a parliamentary session should be presented in a single page, independently of how long it is. However, "Previous/Next" buttons can and should be present to point users to the previous or next session that took place.

What you don't show is as important as what you show

Every visual element has an effective cost. For each graphic entity that is included in a site, it is incurring on an "interface debt", which is reflected on the time necessary to "parse" the site in order to find out the path to the desired destination.

The fact that it doesn't cost a lot to add elements to a page (a few clicks at most) hides the fact that it is in fact an expensive act — screen real estate is limited and every element added will reduce the margin of action to add more.

It is said that a letter is defined not only by its shape, but also by the negative space around it. We can salvage this commonplace notion in order to argue that **empty spaces are necessary** in order to highlight the existing graphic elements.

There is no way to avoid incurring "interface debt", as it is obviously necessary to include a set of graphic cues and entities in order to allow for proper navigation and understanding of a portal's structure. In the same way, the value of parsimony should not be underestimated when the time comes to consider what goes in; what doesn't go in will affect the outcomes as fundamentally as what stays.

3.2 Layout

Visual hierarchy

Hierarchy is key on any layout, and this hierarchy is stipulated by a number of factors: size, positioning, color, all relative to other existing elements. Making the text bigger is probably the easiest and most commonly used technique to draw attention to a particular element. It should be used judiciously, since its overuse will have exactly the opposite effect: by

highlighting everything, nothing is highlighted.

Importance should therefore be carefully set by measuring the relative impact of each element in the page in relation to every other item.

A suggested exercise for this would be paper prototyping; with this approach, it's easier to understand the amount and relative importance of elements in a page, and to establish the desired hierarchy which should then be materialised using graphic techniques: **prominent positioning, a different colour and other tactics to draw the viewer's eyes into what's judged as the most important part of the document.**

Question all defaults

Default options are a given in our use of digital tools. When creating a new document in your text processor, many choices will have been made even before the blank document appears: the sheet size, typeface, text size, text color, paragraph spacing, etc. will be defined according to what's termed default values.

The use of default values, and the adherence to them, usually results on a "boring" outcome, since it's so familiar to other users of the software (which, in the case of word processors, means a significant number of people). How many times have you seen an A4 sheet with Times New Roman text?

One good move to visually differentiate one's work is, therefore, to question the defaults and try to deviate from them whenever possible. Consider for instance the frequency of black text on a white background; neither the background needs to be full 100% white, nor does the text needs to be full black — in fact, lightening the text to a dark gray usually makes the text easier on the eyes.

Our approach can extend beyond the parameters of the web content and be applied to more general elements, such as photos; the "default" is to resort to stock photos, but the general public is well aware of the generic and bland aspect they lend to their context. Would illustrations be a better fit for the site's identity? Maybe so and maybe not: defaults are not inherently weaker than other possible choices.

Likewise, it's worth looking at all the visual elements and assess how well the default works: forms, typefaces, colours, button styles, headers, menus, etc.

This should not be interpreted as an advice to adopt an exuberant attitude of deviating from what looks ordinary; as stated above, the typeface used in body text needs to be a familiar one,

since the readability of text is greatly affected by the familiarity of the letter shapes.

Wasteful elements

Following the posture of questioning defaults that we described earlier, it is worth mentioning a set of graphic elements that ends up as being default on official data sites:

- splash/intro pages;
- welcome sections;
- header banners;
- stock photos;
- oversized logos.

While we do not advocate that all of these elements are superfluous, most of the time they don't have a useful reason to be there, while incurring a significant cost in screen area. The top of a page should be faced as the most valuable area, since it's what a person's eyes will face first. **Any non-vital elements that take up space there are inevitably wasteful.** The opportunity to catch people's attention with information relevant to them is too important to be squandered with superfluous graphic elements.

It bears reminding that data portals have the purpose to showcase the data that they contain, and not as much the organisation behind them; though a simple principle, we find that it's very often ignored in the real world. **Data and information are the centrepieces, and the rest should be a way to highlight them.**

Embrace the grid

Modern web design has adopted grids as the fundamental base to base a site's visual structure on. It is worth mentioning that grids have been present in modern graphic design since the sixties, with the Swiss school pioneered by Emil Ruder and Josef Muller-Brockmann advocating a rational grid-based structure for print based designs.

The modern web is ruled by dynamic grids, with popular front-end frameworks such as [Foundation](#) and [Bootstrap](#) being fundamentally based on them. Additionally, they are a useful way to **create layouts that can fit different screen sizes.**

It is unnecessary to re-invent the wheel when there are already so many shortcuts into achieving clean and practical designs that don't sacrifice the designer's creativity; a grid-based

design is one of those, and there's little reason to try and invent an alternative.

Be careful with re-designs

Re-designing a portal is an initiative that should not be taken lightly. Ideally, the design ought to be reviewed regularly, with the help of usability tests; however, this would require the presence of a dedicated design team, which is seldom the case with data portals.

Therefore, it is necessary to keep in mind that users will be initially disoriented by any change in a portal's structure, so any redesign must take into account the need to enlighten current users as to the new structure; again — ideally, the structure should remain as close as possible to what people know, but new integrations to the portal might make this tough to follow.

In the end, it is important to treat a redesign as a potentially disruptive action, to be taken with an iterative approach (making incremental changes over time), with the help of metrics and user testing.

3.3 Typography

Type plays a role

Computer screens are a very different medium from paper when it comes to reading text. For comfortable reading, especially in long texts, larger text sizes are an immediate way to improve things. Unlike text in print media, screen space is available at no cost of ink or paper, so one can afford to be flexible with text layout.

As a simple entry-point to typographic refinement, it is worth starting by defining a comfortable size for text, from headlines to body text, and make use of whitespace to separate and organize different blocks of content. For mobile layouts, text size and whitespace should be somewhat more condensed.

Testing with different audiences is ideal if it can be afforded, particularly with users of more advanced age, who will often require larger text sizes and therefore set the bar for achieving proper readability across all ages. Even if larger-scale testing is not an option, informal readability tests can be performed with fellow employees or family in seconds ("Can you read this comfortably?").

There are other elements to take into account when laying out text, such as contrast and alignment. Different text elements in a page will have different sizes and styles according to their function and importance. Visual hierarchies are always automatically established whether

there is the concern to define them or not; therefore, one must be careful not to draw attention to less important elements, or hide important information because of a small font size.

Titles, secondary titles, tertiary titles must have significant visual weights, so it is appropriate to define different text sizes for each — we suggest a difference of at least 2 points between each. Titles organise the text and give rhythm to the page, so it's important to apply a font that differs from the body text or, at least, a different weight (bold and/or italic).

Typeface choice

Typeface decisions should be heavily based in the context where they will be applied. The typeface choice might be already made in the form of the site's organisation logo and printed material; nonetheless, we advise restraint when picking typefaces for body text, where readability is vitally more important than graphic finesse or affirming the organisation's identity.

Resources such as [Google Web Fonts](#) are excellent to browse possible typefaces, but it's important not to get carried away: two distinct typefaces are usually enough for a good layout (one for body text and another for titles), whereas more than three is rarely a good option.

Typeface combinations

It is a common and usually effective practice to combine multiple fonts in the same page. However, we would recommend using three different typefaces at most:

- one typeface for text;
- one typeface for titles and highlights;
- one typeface for buttons and/or code blocks.

Combining fonts can be a tricky task and there is always the risk of falling into a visual mess. It is therefore important to keep it simple. If there are no set guidelines, it's appropriate to pick a serif typeface for the main text and a sans-serif typeface for titles or the other way around; in the end, it's about what looks good in the developers' and users' eyes, so this is a point where testing is again desirable.

There are many resources online with suggestions for font pairings that can serve as inspiration or a good base to build on — see the references section for two good resources.

Web typographic standards

A few years ago typeface options for web design were very conditioned and there were only a few fonts to choose from. Nowadays, given the extraordinary advances in web technologies and standards it is possible to use almost any font and to be sure that everyone that accesses a page will be seeing the same font. Take advantage of this improvement and choose a font that particularly suits your message and content. **Using a font that is not a web default is a factor of differentiation.**

The current availability of web fonts now makes it inexcusable to use images to show text; this practice has been common before web fonts became mainstream since it was the only way to show sophisticated headers in custom fonts; however, it has the significant downside to keep out users with special needs who use screen readers to access sites, as well as shutting down the possibility of copying and pasting text.

It is worth remembering that the web already adopted a set of typographic standards and patterns, such as the underlining of hyperlinks; those informal conventions have more importance than what it might seem at first glance, and should be followed unless there is a strong reason to try other approaches.

Get a designer's advice

Type can be like a hard subject to grasp even when there are so many resources at hand — it's easy to get lost in the sea of information. It's not by accident that typography is a specialised field, which is why one might consider looking for the advice of an experienced designer. Nevertheless, following the advice above will go a long way in improving a site's typographic refinement and effectiveness.

3.4 User Interface

Reduce the number of steps from X to Y

The heart of interface design is hiding complexity. Each additional step from the home page to the desired piece of information will progressively dissuade users from browsing. A good interface provides simple paths to go through in order to get where a user wants.

A good first question to ask is: **which are the most relevant and central parts of a site?** The subsequent step is to centre the site's design around them; from there, it will be easier to fit the secondary sections in a non-obstrusive way. This principle has been crystallised into the

informal three-clicks rule¹: all essential information should be accessible in no more than three clicks from the start page. It might be the case that a site's internal structure is too complex to be able to comply with this principle, but the approach of simplifying a site's structure is still a good premise to give users a better handle on what they can find.

Effective search

Websites that contain large amounts of information or archives need to provide the user with interface elements that can aid them in navigating through the content.

A clear structure, tagged content, browsing by date or theme are effective ways to guide the users in finding what they need. However, none of these can replace the function of the site search.

A simple and visible search box is a perennial interface element that is recognized by the majority of users; for many of them, the search box is the default way of browsing. Therefore, it is important to make it visible and accessible in all of a website's individual pages.

An internal search engine is a complex feature to build; however, there are a few out-of-the-box solutions that can be adapted or built upon, doing away with the need to develop something from scratch. DuckDuckGo or Google both provide an embeddable search widget that is straightforward to include in any website; while it introduces the reliance on a third-party service, such a solution is preferable to having no search at all.

Many websites also offer the possibility for advanced search, with extra fields and parameters. Advanced search can be very useful and powerful, but it is extremely easy to end up with a complex search interface that will be hard or impossible to understand for some users. Even when it is usable, most of the time it introduces a roadblock to users' navigation by requiring searches to be specified in very restrictive terms, like using mandatory or interdependent fields. If understanding the parameters available in the **advanced search requires specialised knowledge**, it is almost certain that most of the site's audience won't be able to make use of it. In order to avoid frustration, it is important to always provide a simple search option, one where the user can just type a few terms and get results.

Finally, it's important to keep an eye on the site's metrics to determine whether the search interfaces are being used frequently and correctly, and adjust the interface accordingly.

¹ Zeldman, Jeffrey (2001)

Platform independence

Today, there is a multitude of devices, which people access and browse the Internet through. From desktop computers and laptops to tablets, smartphones, game consoles or TVs, there is a vast variation of screen sizes. This creates a context for design that differs much from design for the printed medium: the layout can no longer be planned for a specific size and ratio (the paper page size), but rather as a fluid structure, adapting itself to best suit the screen size in which it is seen.

Providing alternative layouts for mobile devices improves the user experience of a website — many web interface toolkits provide the basic design conventions for a successful implementation. The basic conventions go from viewport scaling (no scrolling in the horizontal axis) to collapsible menus, bigger links and bigger buttons, for better support in touchscreen devices. These principles pertain to a design perspective now known as "**responsive design**", and there is a wealth of resources available for adopting these directives.

Fixed layout structures should be therefore avoided; a few years ago, when there were few mobile device models in the market, that approach made sense. Today, with a sprawling multitude of brands, devices and formats, the design thinking behind web sites needs an update. Likewise, device-specific formats (such as phone apps) should not be considered a priority: any device that connects to the internet will have a browser, and developing exclusively for the browser will save precious resources in the project's overhead, and provide a unified solution that will work for the large majority of users and devices.

Help text

Institutional websites and online services usually provide instructions and help texts to guide users through their dense content.

Help texts are extremely important for the users, they can serve to explain technical terminology, provide guidance and give answers for common doubts, and are usually placed in a secondary level in relation to the main content. They have a vital function, as many users will only succeed in finding what they needed thanks to the instructions.

Many times, help texts are hidden in small buttons that need to be hovered on or clicked in order to be able to read them; they are often presented in a small text size, away from the content that it refers to; and it's common that the help text is too long. These visual characteristics and placement mean that many users just bypass the information, eventually

getting frustrated and not being able to do what they needed. They might write e-mails to ask for help, but most often they will just give up — the worst outcome possible.

Help texts should thus not be buried. Instead, they ought to be made visible and promoted to first-class elements, alongside the content they are intended to explain. This will go a long way in fulfilling user's demands for clarification.

The copy and tone for these help and instruction texts should be carefully planned. **Short, incisive and to the point snippets are significantly more effective than long explanations**, which might have the opposite effect. Such detailed explanations, if absolutely necessary, can be placed at a specific section like Frequently Asked Questions or How-To pages.

Every element is a promise

User interface is a layer between the content displayed in a page and the user, allowing the user to access and interact with the content.

Every element in a website over the which you can perform an action —be it an internal link, a link to another site, an input box for searching, a form to send feedback— should be visually distinct, guiding the user. Buttons, input boxes, links and other actionable elements should be clearly identifiable at a glance, at risk of making users feel lost if one tries a different look for these elements. It is important to remember that there are already informal conventions as to what elements should look like, and skirting around these conventions is an effective break with the expectations of the users.

Styling text with the same colour used in links can trick the user into thinking that text is also a link, or not realise that a certain snippet is actually a link. A good way to avoid falling into these traps is to keep a cohesive style for all elements that have similar roles or functions. The visual style of these actionable elements should clearly differentiate them from each other.

Every time the user is performing an action, such as using the *searchbox*, submitting a survey or loading a video, it is imperative to provide clear visual feedback — a visual response from the interface to assure the user their action was understood. Such feedback is especially important for tasks that take long to perform; in this case, an estimate of the time it will take to complete an action is strongly encouraged. However, the user shouldn't be overwhelmed with fake loading bars, unnecessary animations that might make the action take even longer to complete, or (worst of all) sound.

Confirmation messages are also very useful in the interaction with users. When information is

submitted in a website, be it through a contact form, a feedback questionnaire or registration process, it is necessary to display a short status message informing the user that their submission was successful (or not), and what can be expected to happen later (such as an e-mail reply).

3.5 Web Technologies

Permalinks and longevity

One of the common problems on the Internet is the longevity of things. Many sites regularly go through updates, remakes or, in the worst case, are deactivated and go offline. This is the reason that, whenever an online source is quoted, there is the requirement to include the retrieval date — examples can be found on Wikipedia references or any bibliographic standard for scientific papers. When planning or undergoing a website update or remake, one of the most important things to have in mind is that links need to stay active after any change, however major or minor. If links change or break, all existing references to the site will become inaccessible, and that is an unacceptable scenario.

The best way to solve this problem is not to face it at all, by **planning a sensible link structure beforehand and treating it as sacred**. This will ensure that outside sources who quote or use the website will continue to work. Otherwise, relevant connections can be lost and many important documents lose their trail in the process. The best rule of thumb is to treat links as permanent entities that cannot change; if a change is required, a specialised technical expert needs to be present to take care of the migration issues that will arise.

Sysadmins are crucial

A common occurrence in institutional websites and portals is the need to include the *www* in the URL, failing if it is not present. The prefix *www*² is an artifact from the early days of the Internet and its use has declined in recent years. Nowadays, it is a good practice to make a website accessible with both the "www" prefix and without it.

The details of how to technically solve this situation are beyond the scope of this report, as they involve insight on the technical workings of servers and DNS configuration. Resorting to the services of a proficient system administrator (*sysadmin*) who can be in charge of the technical infrastructure of the website and give it some attentive grooming. Given the

² Wikipedia article World Wide Web on WWW prefix
https://en.wikipedia.org/wiki/World_Wide_Web#WWW_prefix

particular nature of these subjects this is not the kind of task that can be accumulated by a designer but should instead be the sole occupation of a proficient *sysadmin*.

It is a good investment to find a good *sysadmin* or *sysadmin* team, treat them as first-class elements in the project, and acknowledge their seldom-visible efforts to keep the website running smoothly.

Social networks

A social network presence is a good way to outsource the news and updates section of a data portal, saving the time required to build an extra component, and taking advantage of already existing networks.

However, it is unwise to exaggerate the social network angle, lest the portal become an unpaid advertisement for those sites — particularly if the data portal belongs to a public institution. Links and connections to them should be placed where it matters: it is acceptable and encouraged to have "Share" buttons, but less so to have a sizable banner in the front page with the list of social networks that an official data portal is at.

A good way to **avoid giving too much focus to external services**, such as social networks, is to make custom buttons instead of using the one that are provided by those services. A designer can take care of adapting and designing a set of icons or buttons that are recognizable but that blend in well with the overall style of the website.

Proprietary technologies vs. standards

There are many frameworks and web technologies available when it comes to making a website or portal. Given this scenario, the quest for the best option is not always an easy one.

We will not propose any particular solution, as each data portal will have different features and different needs. We do emphatically recommend the choice of standard, **open Web technologies and frameworks whenever possible**. Standard technologies are a way to assure the longevity of a website, as they facilitate backward compatibility and feature vibrant support communities and forums.

Proprietary solutions such as Adobe Flash or Microsoft ASP.net can seem appropriate in the short run, but will make the website more dependent on the company's support (often requiring significant investment). There are, on the other hand, a number of documented success cases of public institutions open sourcing their platforms' code.

Follow your metrics to see what the problem spots in your site

Using site analytics is a common sense move in modern Web development. Understanding the paths that users take through the site can be key in pinpointing stumbling blocks in the site's structure. It is worth devoting resources to analysing a portal's metrics in order to find avenues for improvement on the site's interface, many of which user testing might not be able to highlight.

[Google Analytics](#) is the most popular choice for this. Google's tracking and information mining can be considered an issue, though, when the time comes to choose an analytics package. There is a fully-featured open source alternative, [Piwik](#), available for consulting all kinds of metrics.

4 Conclusions and recommendations

Data portals are specific in that they are very easy to get wrong, either by not giving users enough clues and tools to allow them to autonomously navigate the site, or by not being able to effectively manage the inherent complexity of their contents.

We have so far approached the essential concerns that good interface planning can be tackled. The engagement generated by a data portal is indeed its lifeline, and there are subtle ways that this engagement can be fostered or, alternatively, hampered. We hope that this report succeeds in highlighting the need for taking interface design principles into account, without needing to include a specialised design team into a project — which nevertheless is a good move if the project's budget can accommodate it.

There are other concerns beyond design that fall out of the scope of this report. One of them is having a good systems administration team, as described above, who can deal with the hidden but very real infrastructure. The other concern we would highlight is copywriting. Having a dedicated **copywriter who can review the existing text and set a consistent tone across the site is one of the best moves to improve the quality of a data portal**, since the accessibility of such sites can be jeopardised by carelessly edited copy text. Additionally, having a person separate from the development team review the texts will help spot places where specialist terminology is used needlessly, as well as occasions where reformulation is needed to cater to non-specialist users.

It is our hope that this document will in some way help and enlighten the efforts of project leaders, developers and technical staff in charge of data portal planning, development and maintenance. As a closing note, we remind that this document is distributed under a CC Attribution license, meaning that its distribution and improvement are not only allowed but encouraged; the authors would sincerely welcome any kind of feedback and improvement suggestions.

It's too easy to focus and get lost in design details, losing sight from the greater goal of providing good tools to empower people with access to relevant information. Ideally, planning the back-end architecture and front-end design should be a simultaneous process, but not usually possible.

A good resource website using default themes – or no styling at all – is a much better starting point than a cleverly designed site with an incomplete backend or database (we made this

mistake). **The data needs to be ready at launch, whereas the design can be improved as you go.**

These are just pointers into how design principles can be considered in the goal of providing an interface that works in catching regular users' interest. Front-end design can be a very strong tool in turning a simple data service into a relevant and attractive resource that can be used by audiences that you wouldn't even consider in the first place.

5 References and resources

Typography:

- [A List Apart](http://www.alistapart.com/topics/design/typography) is a web and printed publication that specialises in web and interface design. <http://www.alistapart.com/topics/design/typography>
- [Hand-Picked Tales from Æsop's Fables with Hand-Picked Type from Google Fonts](http://femmebot.github.io/google-type) is an inspiring showcase for combinations of typefaces made available through [Google Fonts](https://www.google.com/fonts). <http://femmebot.github.io/google-type>
- [Google Fonts](https://www.google.com/fonts) provides pairings suggestions in the individual font page. See an example for pairings for [Open Sans](https://www.google.com/fonts/specimen/Open+Sans). [http://www.google.com/fonts/specimen/Open+Sans - pairings](http://www.google.com/fonts/specimen/Open+Sans_-_pairings)

Interface Design for the web:

- Krug, S. (2001). [Don't make me think](#): New Riders Publishing.
- Tidwell, J. (2010). [Designing Interfaces](#): O'Reilly Media.
- Zeldman, J. (2001). [Taking Your Talent to the Web](#): New Riders Publishing.

About the Authors

Ana Isabel Carvalho and **Ricardo Lafuente** are design and interface specialists who focus on interactions between the open web, public data and design research; they work between the fields of print, digital media and engagement with data and software.

They head the Porto office of the [Journalism++](#) data-driven journalism agency, founded the [Manufactura Independente](#) design studio, and coordinate the activities of [Transparência Hackday](#), Portugal's most active **open** data collective.

Their recent endeavours include the design, implementation and operation of the parliamentary information portal [Demo.cratica.org](#), as well as the development and maintenance of [Central de Dados](#), an independent data portal.

Copyright information

© 2015 European PSI Platform – This document and all material therein has been compiled with great care. However, the author, editor and/or publisher and/or any party within the European PSI Platform or its predecessor projects the ePSIplus Network project or ePSINet consortium cannot be held liable in any way for the consequences of using the content of this document and/or any material referenced therein. This report has been published under the auspices of the European Public Sector information Platform.



The report may be reproduced providing acknowledgement is made to the European Public Sector Information (ePSI) Platform.