

Fortuity Research Report 6

There are two reports from the boffins at Fortuity Research this time around: one on web forms and another on pattern matching search techniques.

Forms on the Web

How many forms cross your desk every day? How many of them are poorly worded or badly designed, with ambiguous questions or insufficient space for the answers? What proportion of them get a response, and how many end up as landfill every year? How many do you send out, and how much does it cost? How long does it take to enter the answers into a computer?

Bad form design used to be like the weather – everyone complained but nobody did anything about it; but now the rise of web-based forms means that designers can take the time and trouble to get things right, and whole forests can be saved. Fortuity Labs' form expert – recruited from the racing pages of a major newspaper – has looked into some of the form design systems available via the web, and compiled the following report:

EmailMeForm

EmailMeForm (www.emailmeform.com) is an online form design assistant. It takes the designer through seven steps and produces the form as HTML code, which can then be hosted by EmailMeForm or added to a page on your own website or intranet. The form includes optional validation to check that email addresses, for instance, are in the proper format. The procedure for specifying multiple choices is a little cumbersome, but works reasonably well, and computer-savvy users can go on to add their own formatted HTML code and CSS styles to the form. Other options include check boxes and the capacity for the user to upload and submit a file. An optional 'Captcha' copy-these-letters box can be included as a way of blocking automated spam. Existing forms can be viewed, copied or edited by the designer at any time.

User data entered into the form is processed by PHP files on the EmailMeForm server and then sent to the designer by email. Uploaded files are sent as attachments. A copy can also be sent to the person submitting the form, provided they have included an email address. Data is laid out on a line-by-line basis but can also appear in comma-separated format for copying into Excel or other analysis programs. On completing the form the user is passed on to a URL address which the designer has specified – normally a 'thank you' page.

EmailMeForm designers need to register with an email address and a password. The basic service is free and allows up to 5 forms and 50 fields. A paid upgrade for \$US10 per month removes these limits and adds a few extra features, including the ability to accumulate multiple responses and download them as a single file, and copies of the PHP processing files which can then be stored and used offline.

The verdict? Quick, simple, and works as advertised. But as with any online storage and processing facility, designers should be aware of security and confidentiality issues.

[Screenshot – EmailMeForm - testing a form design.gif]

FormLogix

FormLogix (www.formlogix.com) is more ambitious, offering some pre-built form templates and the option to extend a form over several webpages. There is a public trial option, but to save and re-use forms, designers will have to register for a thirty-day free trial. Unlike the step-by-step approach of EmailMeForm, FormLogix brings up the entire form at once in a JavaScript workspace with many toolbar buttons and options. At first this is a little confusing, but perseverance will pay off and reveal many formatting and control features. A help system provides assistance in simple, if occasionally ungrammatical, English.

Forms are stored on the FormLogix website and accessed via a URL. Unfortunately, to fill in a form users must also be registered with FormLogix, which reduces its usefulness considerably. The information submitted by respondents is stored on the site and can be accessed by the designer via their account. It can be exported in a wide range of formats, including Excel, PDF, CSV and even Google Maps for geographic data.

Form designers who want to continue with FormLogix have a range of payment options – per form, by submission numbers, or by time. A six-month subscription is about \$US30.

Verdict: Rather overwhelming for beginners. The need for respondents to register and supply an email address also made me a little suspicious about the site's motives: are they selling addresses to spammers? But a very flexible system for dedicated designers.

[Screenshot – FormLogix - form design screen.gif]

FormSpring

FormSpring (www.formspring.com) provides a free trial account allowing designers to create and offer up to three forms; paid options range from \$US14 to \$US159 per month for up to 1,000 forms with 100,000 submissions each – enough to mount your own Australian Census five times over. There is live support for paid accounts, as well as a free FAQ and a blog.

The FormSpring interface is clean and there is a four-step approach to creating a form. Designers can start with a blank, choose a template, or upload an existing HTML form to work with. Forms can be divided into sections, and fields can be shown or hidden on the basis of other field entries, so a respondent under 18, say, can be given a different set of options to an adult respondent. Predefined field clusters are provided for common information like street addresses. There are date and time fields, sliders for numeric values, and other well-thought-out options like randomization for drop-down lists to reduce selection bias. Forms can also include formatted text and graphics and questionnaire-style response scales. Form extras include layout controls, progress indicators for multi-page forms and (on paid accounts) 'Captcha' boxes to limit spam. Hidden fields can record submission times, IP numbers and the type of browser used.

In Step 3 designers can set a time zone and a template, and choose whether or not to store (as well as email) the respondents' data. Filtering can be used so that only responses matching certain values are sent on. Paid accounts also provide additional security and user verification options. Paid account holders can also create and use their own form templates.

The completed form is hosted on the FormSpring site – which is accessible to everyone – but can also be downloaded and used on your own website or intranet. Stored data can be downloaded in Excel or CSV format, and there is an option to filter by submission date, so

that only newer records are downloaded.

Verdict: Very impressive. If there's anything that FormSpring has left out it certainly isn't obvious. The step-by-step approach makes design relatively easy, while building up tremendous capabilities. Ideal for anything more demanding than the most transient data collection.

[Screenshot – Formspring - form design screen.gif]

Web-Form-Buddy

Web-Form-Buddy (www.smallgreentree.net) doesn't provide hosting, but they allow you to create forms to be hosted on your own website. The process is quick, but the demo form is extremely basic and must be uploaded to a specific location via FTP. I couldn't find any way to customise a form on the site other than by manually rewriting the HTML code, so I'm not sure what this site is really offering, other than access to their PHP scripts. There is a 14-day free trial, after which a subscription is \$US40 per year.

Verdict: a good beginning which doesn't seem to go very far.

Freedback

Freedback (www.freedback.com) uses a similar 'building' approach to FormSpring, but its options are more limited and it doesn't host the forms itself, simply providing the results as an HTML file. There is a 14-day trial period, after which it costs \$US9 per month for a basic one-form account.

Verdict: OK for an economy option, but a little extra per month buys a lot more at EmailMeForm or FormSpring. For the moment, these are the front-runners and Cup favourites.

ERE pattern matching

"How do you attain such astounding productivity in your search routines?" the Fortuity Linux programmers hope that someone will ask them at a party one day – assuming they ever get invited to a party – because they have an answer all ready: "By using pattern matching with regular expressions." But since this is the kind of phrase that sends party-goers off with a glazed look in search of another drink, they've written a report instead to try and show what all the fuss is about.

Veteran PC users may remember the days of command-line operations, when a user could tack wildcard characters on to a command to specify a particular group of files: thus

```
del *.txt
```

would delete all files with .txt extensions, and

```
cp *.do? A:
```

would copy all files with extensions like .doc, .dot and .dog to the A: drive.

The POSIX Extended Regular Expression (ERE) standard represents a leap forward from this simple approach, and provides a flexible and easily-customised approach to text searching. It's widely used within Linux and other open source software such as OpenOffice,

but is at least partly implemented in many Windows-based programs such as Microsoft Access and Word. It's also an integral part of batch file programming on all platforms: Linux, Windows and Macintosh. The exact characters used differ slightly from one platform to another, but the same functionality remains available.

Finding strings with ERE

ERE works on two levels, finding and replacement. Using it to search within a word processing program like OpenOffice Writer requires the More Search Options panel to be visible and a 'regular expression' checkbox to be ticked. You will usually want case sensitivity turned on also. The search string can then be written in such a way as to match many different strings at once. Some of the more common ERE conventions are:

^Bob – A preceding carat character means 'look at the beginnings of paragraphs only'. So this term will find 'Bob' (and 'Bobette') only when they appear at the beginning of a para.

Sue\$ – Finds the specified characters only when they occur at the end of a paragraph.

sue\> – Finds 'sue' at the end of a word – e.g. 'issue' – but not in the middle of a word – 'tissues'. The backslash-less-than combination searches the beginning of words.

B[oi]b – any character within square brackets can be used in matching; so this will find 'Bob', 'Bab' and 'Bib'.

S[a-z]m – where ASCII defines a consecutive range of characters, all these can be indicated by listing the first and last, separated by a hyphen; so this will find 'Sam', 'Sbm', 'Scm', etc.

S.m – a period stands for any one character, like the old DOS question mark – 'Sam', 'S2m', 'S%m', etc.

Che+se – the plus character finds patterns where the character preceding the plus sign appears one or more times – 'Chese', 'Cheese' and 'Cheeese', etc.

*Che*se* – the ERE asterisk does almost the same, but the preceding character can also appear *no* times – 'Chse', 'Chese', 'Cheese', etc.

Che{2}se – finds patterns where the preceding character occurs twice only – e.g. 'Cheese'. Can be used in combination with other codes: thus *Ch[ae]{2}se* finds 'Cheese', 'Chaese' and 'Chaase' but not 'Chaeeese' or 'Chase'.

Che{2,4}se – as above, but the preceding character can occur between two and four times.

Characters with special meanings in ERE can be searched for by preceding them with a backslash: thus

\.) – searches for a period before a closing parenthesis, and ** searches for a backslash.

ERE also defines special identifiers for finding tabs, carriage returns, digits, white space, and the start and end of words.

Replacing strings with ERE

ERE has some features specifically for text replacement.

& – an ampersand duplicates the text that was sought for; so looking for 'oak' and replacing it with '& tree' would produce 'oak tree'.

A search string can be broken down into numbered sections with parentheses, and rearranged by reshuffling the sections in the replace field; e.g. searching for '(Little) (Red) (Riding) (Hood)' and replacing it with '\$4 \$2 \$3 \$1' will give 'Hood Red Riding Little'.

Assume for example that you have a ninety-page text file where some paragraphs start with a person's name, and you want to swap this around so that 'Joan Smith' becomes 'Smith, Joan' – while any non-name and any name *not* at the beginning of a line remains untouched. One pattern-matching search will do it:

Search for: `^([A-Z][a-z]+) ([A-Z][a-z]+)` – Find any two words at the beginning of a paragraph which both begin with capital letters.

Replace with: `$2, $1` – Swap the first word with the second and add a comma between them.

Another example: assume you want to find all valid Visa Card numbers in a document.

Search for: `4[0-9]{12}([0-9]{3})*\|>` – this will find strings of thirteen digits, starting with a four, and optionally followed by a colon and three more digits. Strings of fourteen, fifteen, eighteen or twenty digits will be ignored.

The power of it! and this is why ERE in one form or another is making its way into all kinds of text-manipulation programs and systems. Consult the Help system on your word processor, or search for 'regular expressions' in Wikipedia for more details.

June 10, 2012