What on Earth is Google CL

As Google breaks from the shackles of the web browser, Bob Moss explains its new command line tool – bringing Google Docs to a shell near you.

Hang on, you’ve covered WebM, Chrome OS and Android in past issues and now you’re writing about Google CL. Is this magazine turning into Google Format?

No, but Google has been releasing a lot of open source and Linux-related products of late. Having a huge, all-conquering web giant developing products and investing in everybody’s favourite free operating system will benefit the wider Linux ecosystem, which is why these products receive so much coverage.

Alright, that makes sense. So tell me, what exactly is Google CL?

Google CL (or Google Command Line to use its full name) is a package that you can install on absolutely any Linux distribution, which enables you to execute a wide variety of operations using various Google services. Whether it’s uploading a photo, writing a post for your Blogger account or adding a new event to your calendar, you can do this with the use of just one line in the terminal, or automate those actions in a shell script.

Why did Google develop this? Surely this must have taken lots of time for a pretty niche tool?

As it happens Google CL was originally created by Google for use by its employees. However, because it’s so useful Google has released it on its own Google Code project hosting service and made this freely available to anyone and everyone to use, develop with and improve.

That sounds awesome! Which services are supported?

You will be glad to know that Google Calendar, Google Docs and your Gmail contacts can all be managed from the command line using this tool. With one line, such as:

```
google calendar add “Meeting at Pilkington tomorrow noon”
```

you can add events to your calendar. You can do similar things with your files stored in Google Docs and with the contacts that you’re probably syncing with the Gmail service.

That’s fantastic! I guess Gmail is supported too?

Unfortunately Google’s webmail service is not supported, but most mail clients that are native to Linux tend to be compatible in some way with the command line interface, so in theory it is not difficult to set something of the ilk up yourself. Mutt and Alpine are better suited to this as they are text-mode anyway, but popular mail clients such as Evolution and its associated background daemons can be invoked in the terminal or via a shell script to send emails or check for unread arrivals.

OK, so what about the rest of the supported services?

Blogger is supported, so you will be able to post items to your blog in a similar fashion to adding entries to the previous services. An example command would be:

```
google blogger post --blog “Personal Blog” --tags “linux, examples, fun” post_text.html
```

Here we select the relevant blog, apply tags to the post then grab a HTML file containing the relevant content. This can contain video and photo content, but these will need to be hosted elsewhere as at the time of writing these were not imported with the text.

But I’m not all about text. What about other content, like videos and photos?

YouTube support is also provided for video uploads and Picasa provides photo and picture upload functionality. To create an album and import a folder of pictures to Picasa you could use something similar to the following:

```
google picasa create --title “Cool album” ~/path/to/cool/album.jpg
```

How far can I customise what I post?

Obviously a command line tool is not going to ‘view’ content online (though you can do this using Lynx, a text-mode web browser) but you can quickly and easily create a post for Blogger complete with a title, subject, tags and the content that will go in the body section. With YouTube you can upload videos as long as they are in the right format with a title as usual, and Picasa also allows photo uploads with tags.

With all this potential to use each service, can I do cool stuff?

Indeed you can. A shell script is any file with a .sh extension that contains terminal commands. The first command to be executed is at the top while the last command to run is at the bottom, and your script can even take parameters (referenced as $n, where $n is the index of the parameter).

Using a shell script you could in theory automate the posting of video and photo content and then create a blog post that links to them within some template text. You could create a task in Google Calendar every time you create a new To-Do item, or even something as trivial as creating a new event every month until you disable the script. In short, this simple tool can be used in almost any way to make the best use of your Google account. Subscribers will find more details on shell scripting and scheduling with cron in the tutorial we provided in LXF126.

I love Linux but I dual-boot with Windows. Is Google CL cross-platform?

Google CL is provided as a Deb package that you can use with Ubuntu and Debian-based distros, and as a source package for other distros. It also requires gdata-python-client as an additional dependency. To run Google CL on Windows however you need to have the Google

LXF136 October 2010

www.linuxformat.com
Data API and Python already installed on your system, then follow the guide at this blog post: http://bit.ly/bLMaKF.

» So does this mean my existing Linux applications will work better with Google services?
That is a possibility. At the time of writing Google CL was only just released, but by the time you read this you may well see the package maintained in your distribution repository and counted as a dependency for many common desktop applications such as photo managers, mail clients, blogging clients etc. Parts of the Google CL code may have even been included natively in the application source code, but only time will tell how niche or mainstream this handy tool becomes.

» ... and we could see more Linux apps working like a front-end to this?
Again, we’re still speculating, but it would be trivial to write a GTK or Qt 4 front-end for it, and as we have already discussed you can create a shell script and use crontab to take advantage of this package. You can use Google CL to manually submit content to the web without firing up your browser or to automate and schedule your actions, meaning that this simple tool is highly flexible and powerful.

» I can’t wait to get going with this!
Where can I get more information?
You can check out the project homepage at http://code.google.com/p/googlecl, which contains an extensive wiki, a list of any bugs and the package downloads. The introductory blog post at http://google-opensource.blogspot.com/2010/06/introducing-google-command-line-tool.html provides a concise introduction to Google CL as well as a couple of examples to try out. [EU]