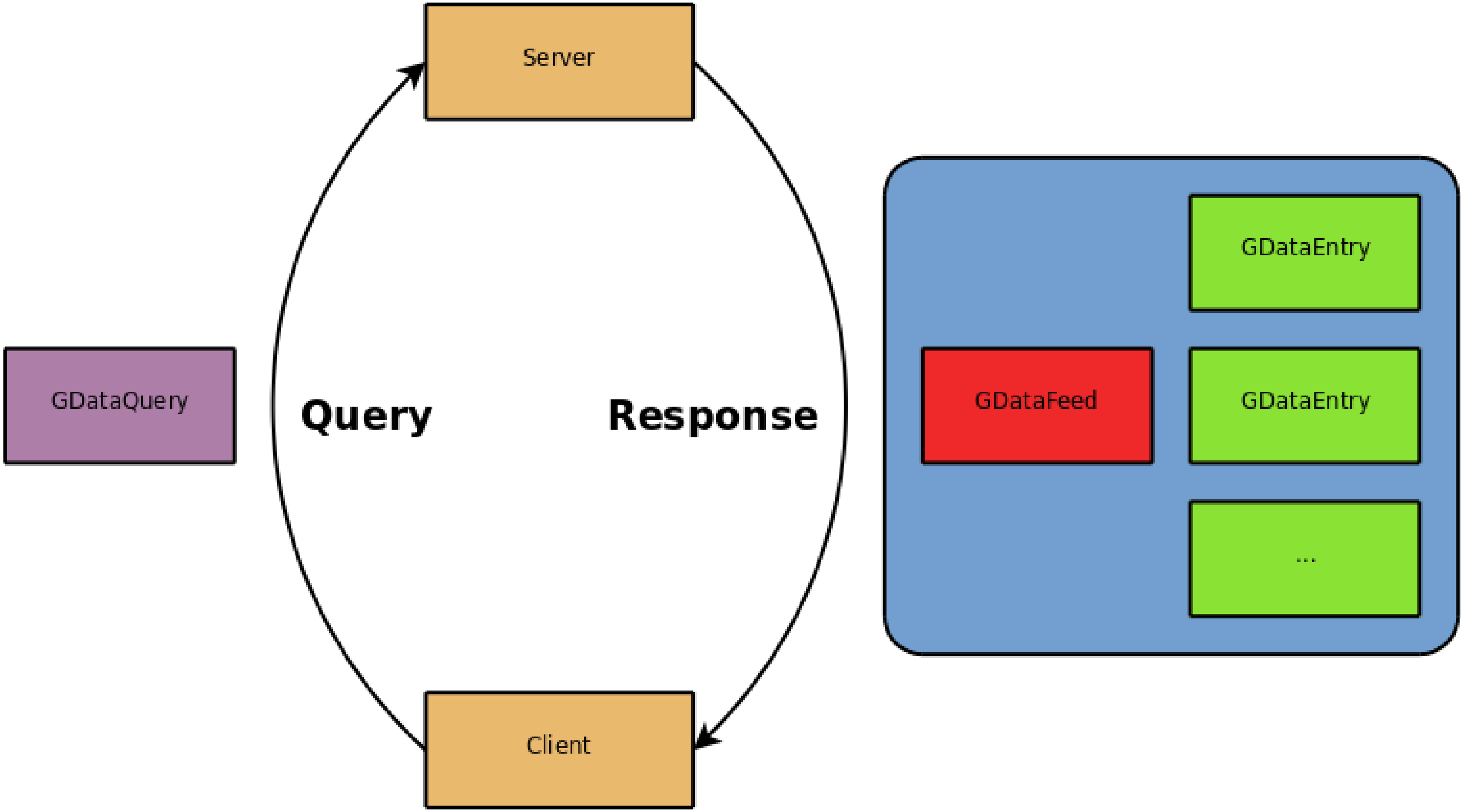
libgdata and web integration

Philip Withnall, 2009-07-04

GData protocol



GNOMEiness

- libgdata is GNOMEy!
- Depends on libsoup, libxml, GLib and GIO
- Makes extensive use of GObject
- Alternative GData library for those who aren't GLib fetishists: libgcal
  - Depends on libxml and libcurl
  - Supports Google Calendar and Google Contacts

Code!

```
GDataYouTubeService *service;
GDataFeed *feed;
GDataYouTubeQuery *query;
GList *entries, *entry;

service = gdata_youtube_service_new (DEVELOPER_KEY, CLIENT_ID);
gdata_service_authenticate (service, USERNAME, PASSWORD, NULL, NULL);

query = gdata_youtube_query_new (NULL);
gdata_youtube_query_set_location (query, 27.9, -15.6, 30000.0, FALSE);

feed = gdata_youtube_service_query_videos (service, query, NULL, NULL, NULL, NULL);
entries = gdata_feed_get_entries (feed);

for (entry = entries; entry != NULL; entry = entry->next) {
        GDataYouTubeVideo *video = entry->data;
        g_message ("%s: %s", gdata_entry_get_title (video),
                             gdata_youtube_video_get_player_uri (video));
}
```

Web integration

We already have…

- YouTube in Totem
- Mail in Evolution
- Calendar in Evolution
- PicasaWeb in F-Spot
- Documents in Nautilus (in progress)

We could do with…

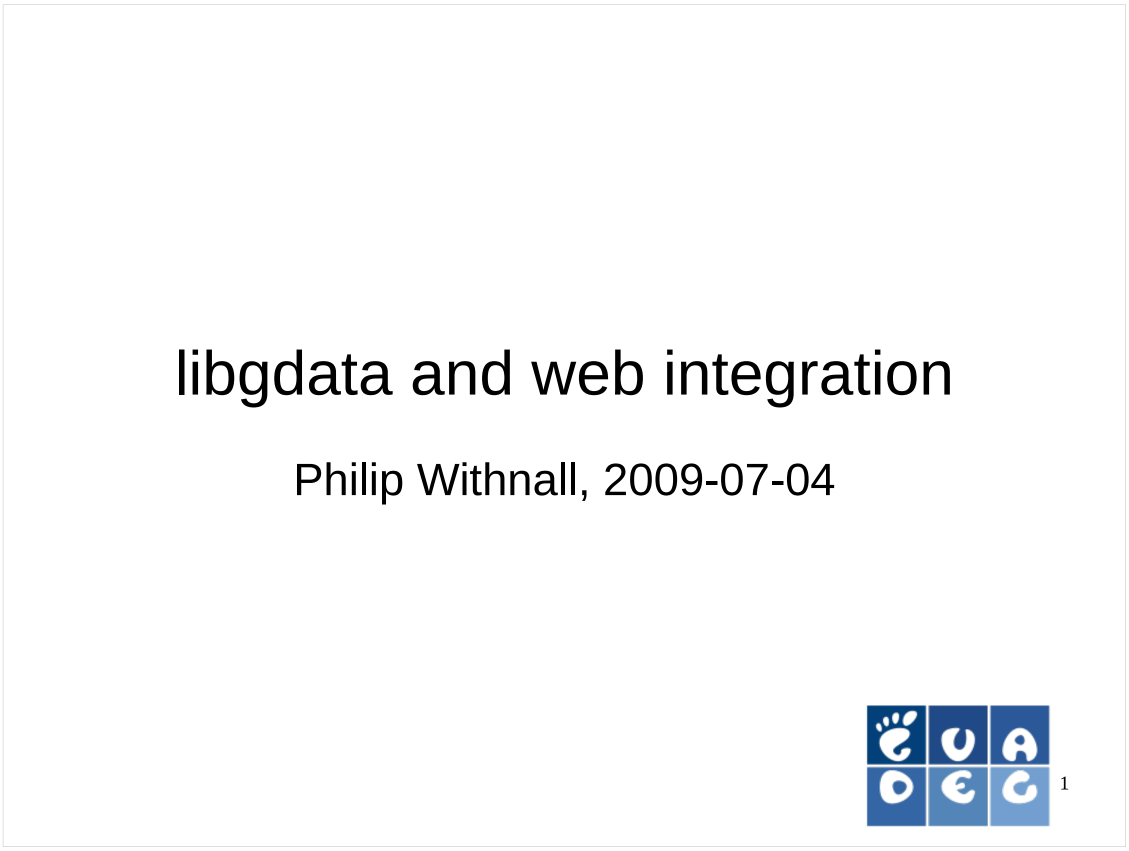
- YouTube in PiTiVi
- Documents in AbiWord and OpenOffice
- *Insert your project here*

Openness

- libgdata is licenced under LGPL 2.1
- GData protocol is openly-documented, but not open-source
- Should we be encouraging use of non-open services?
- Opposition to inclusion of libgdata in GNOME's desktop release suite

Links

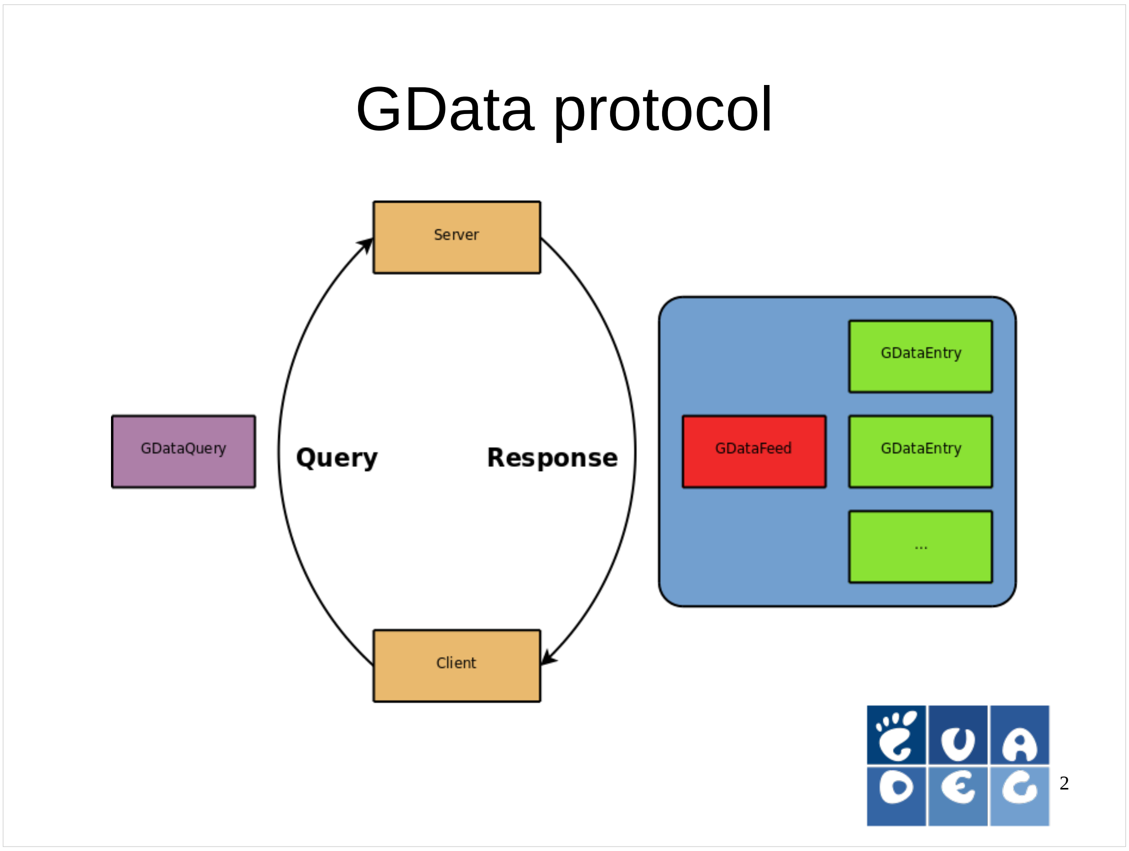
- libgdata: `http://live.gnome.org/libgdata`
- libgcal: `http://code.google.com/p/libgcal/`
- GData protocol: `http://code.google.com/apis/gdata/overview.html`



libgdata and web integration

Philip Withnall, 2009-07-04

I'm going to briefly introduce you to libgdata, my latest project. It's a library to access services using the GData protocol — i.e., Google services.



GData protocol

Server

Query

Response

Client

GDataQuery

GDataFeed

GDataEntry

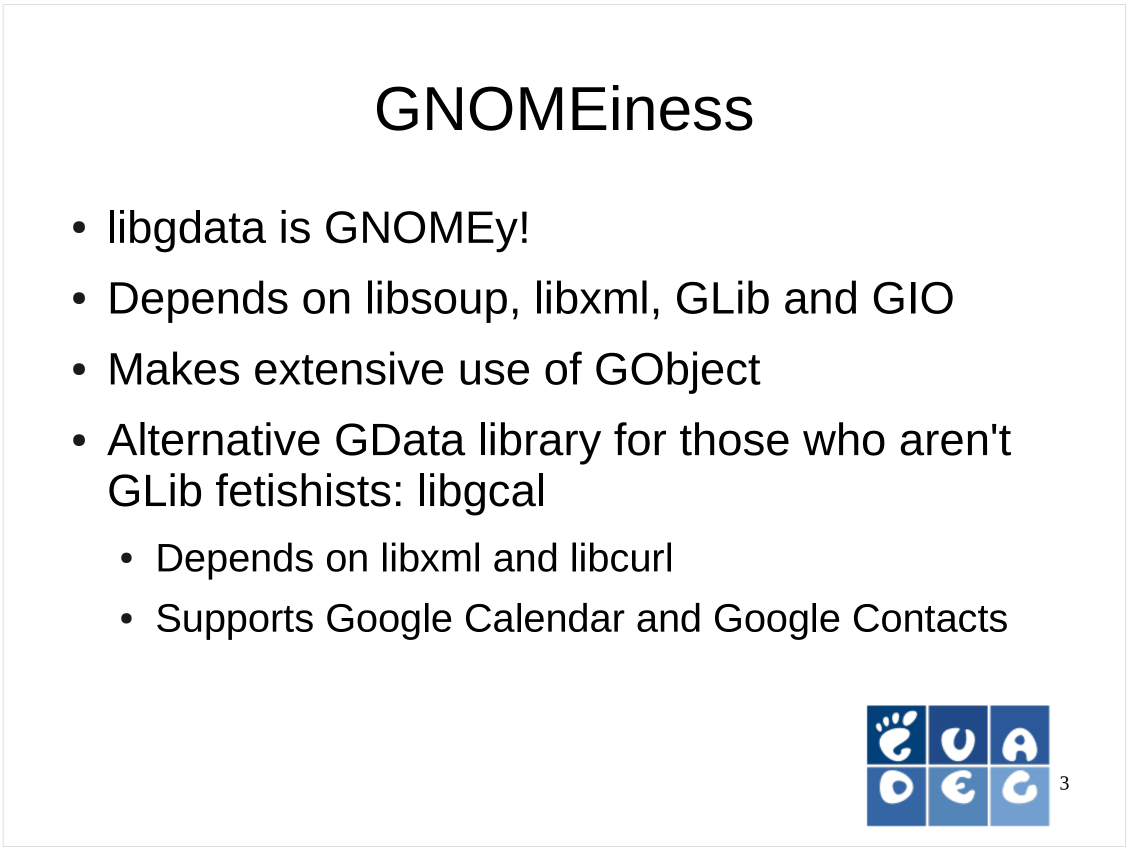
GDataEntry

...

GData is a fairly simple data transfer protocol designed by Google for use with their products. It's based on open standards like the Atom Publishing Protocol. For a typical query operation, the client will send the server an HTTP request with the query parameters encoded in the URI, and will receive an Atom feed in response. Each entry in the Atom feed will be one of the results — an event from your Google Calendar, for example.

In libgdata, a query is represented by GDataQuery (or a subclass); a results feed is represented by GDataFeed, and an entry by GdataEntry.

While GData is the core protocol, each of Google's services have their own namespaced additions on top of it, represented by libgdata by a subclass of GDataService (a singleton which represents your connection to the server).

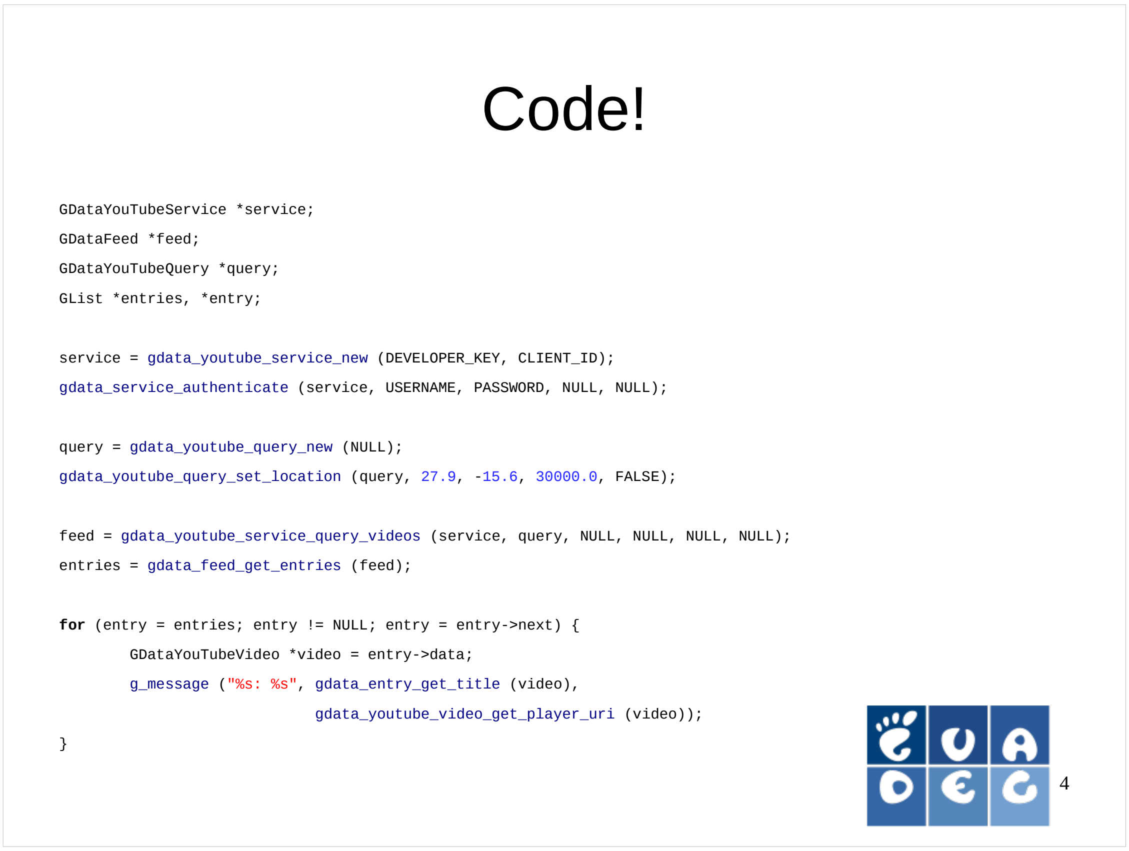


GNOMEiness

- libgdata is GNOMEy!
- Depends on libsoup, libxml, GLib and GIO
- Makes extensive use of GObject
- Alternative GData library for those who aren't GLib fetishists: libgcal
  - Depends on libxml and libcurl
  - Supports Google Calendar and Google Contacts

That's how the GData protocol works, so now let's now look at libgdata. Firstly, it's GNOME through-and-through. It's highly object-oriented using GObject, and drags in such dependencies as GIO.

If you're not a GLib fetishist, there is another, unaffiliated, library called libgcal which provides much the same functionality, which currently has support for Google Calendar and Contacts.

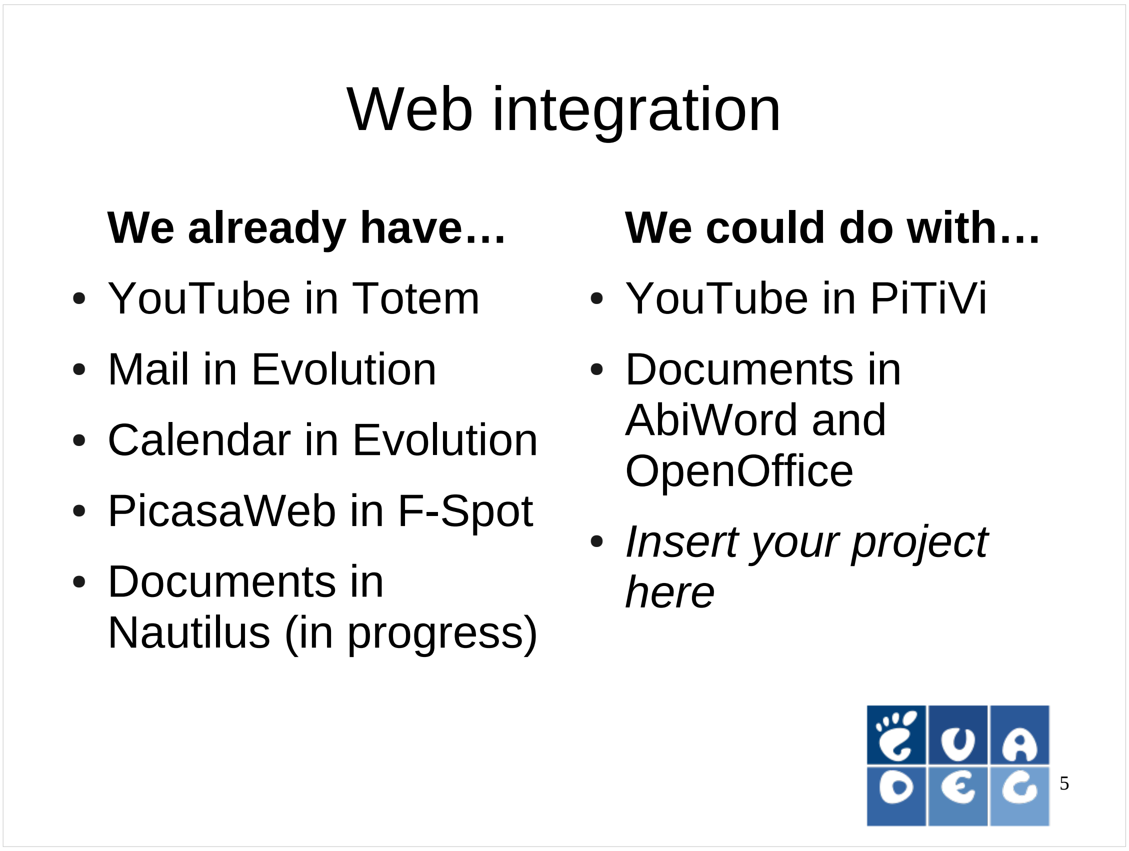


Code!

```
GDataYouTubeService *service;
GDataFeed *feed;
GDataYouTubeQuery *query;
GList *entries, *entry;

service = gdata_youtube_service_new (DEVELOPER_KEY, CLIENT_ID);
gdata_service_authenticate (service, USERNAME, PASSWORD, NULL, NULL);

query = gdata_youtube_query_new (NULL);
gdata_youtube_query_set_location (query, 27.9, -15.6, 30000.0, FALSE);

feed = gdata_youtube_service_query_videos (service, query, NULL, NULL, NULL, NULL);
entries = gdata_feed_get_entries (feed);

for (entry = entries; entry != NULL; entry = entry->next) {
        GDataYouTubeVideo *video = entry->data;
        g_message ("%s: %s", gdata_entry_get_title (video),
                             gdata_youtube_video_get_player_uri (video));
}
```

Here's a brief example of some code using libgdata. This will query YouTube for all the videos geotagged as being on Gran Canaria, then print out the video name and the player URI.

Note that all the typecasting, error checking and memory management code is missing here for clarity's sake. With proper error handling, this code would be about twice as long, since each network operation could fail.



Web integration

We already have…

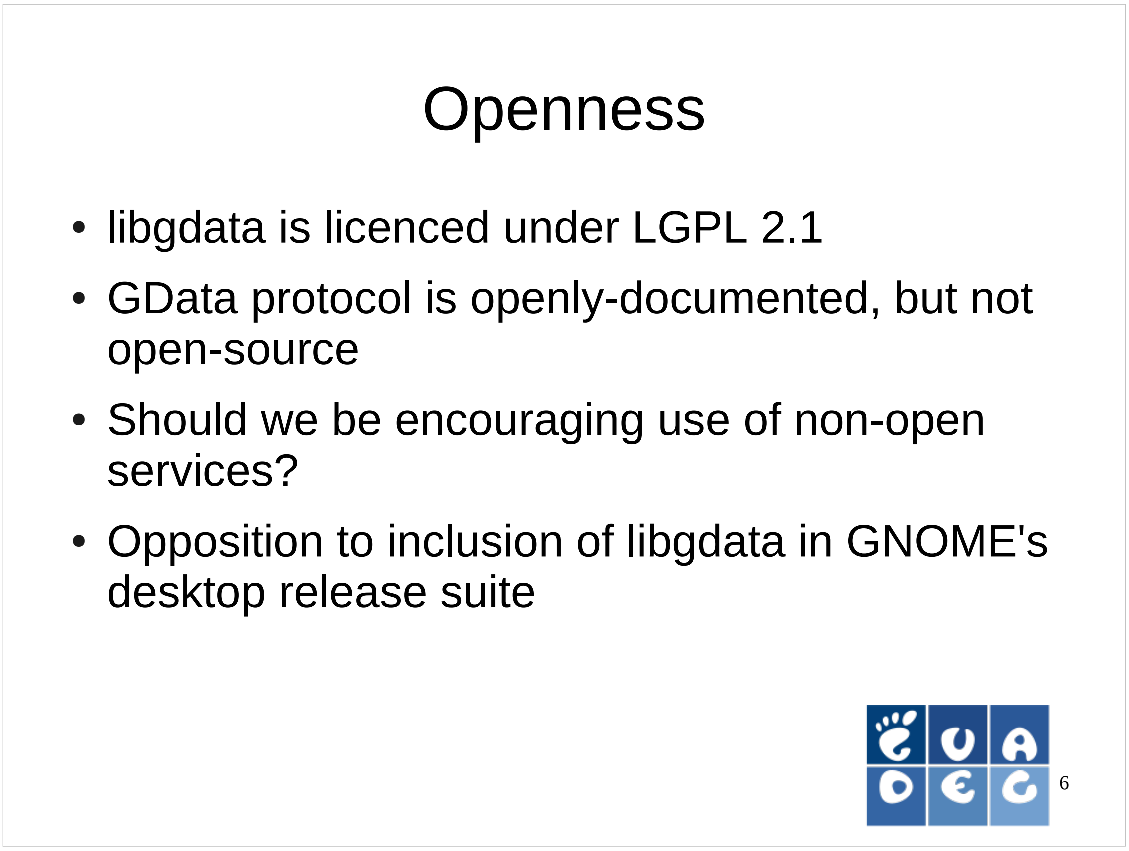
- YouTube in Totem
- Mail in Evolution
- Calendar in Evolution
- PicasaWeb in F-Spot
- Documents in Nautilus (in progress)

We could do with…

- YouTube in PiTiVi
- Documents in AbiWord and OpenOffice
- *Insert your project here*

This is normally the point where the presenter reveals their master plan for taking over the world, or integrating the web with the desktop. Unfortunately, I have no such plan. Currently, I'm focussing on integrating web services where appropriate and useful in the desktop. We have a YouTube query plugin in Totem, Mail support in Evolution (though that's not with libgdata), etc.. Google Documents support in Nautilus is being worked on by Thibault Saunier as part of this summer's GsoC.

What else can we do? I've come up with a few ideas here, but it's up to each application maintainer to see how best to integrate web services (and not just Google's) in their application.

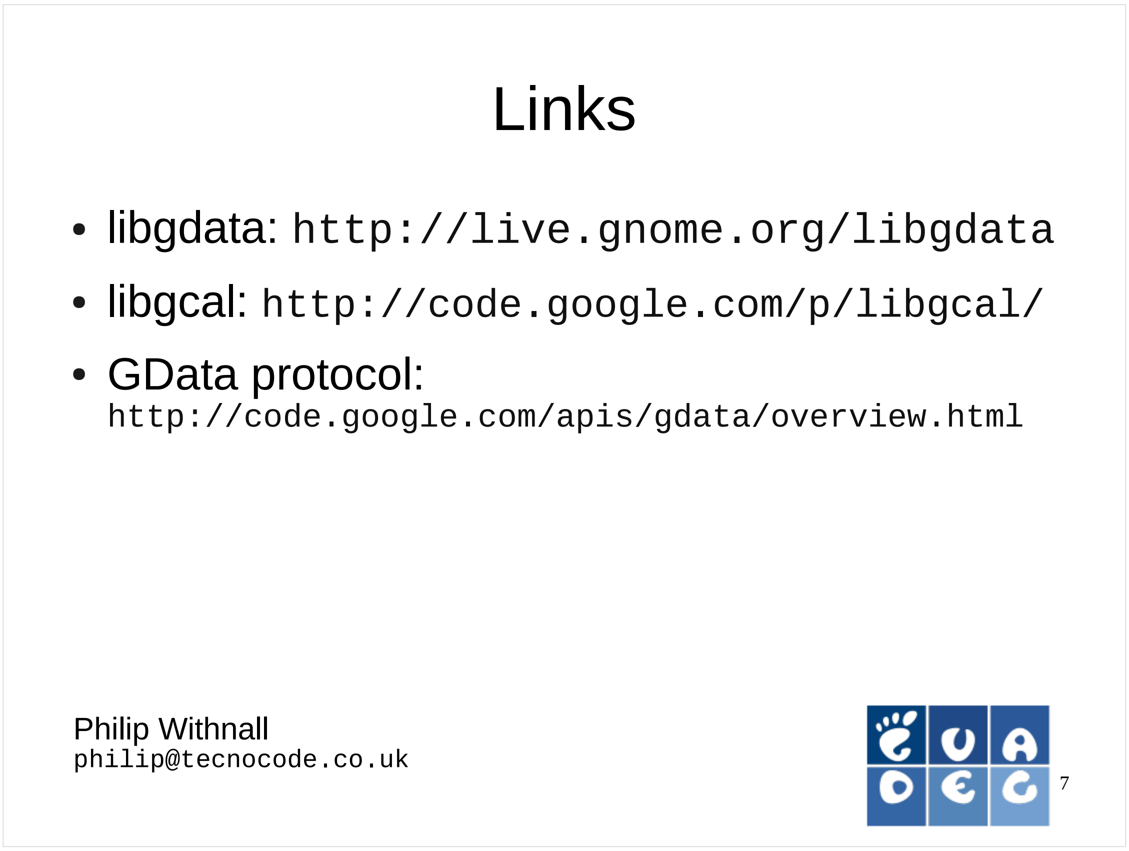


Openness

- libgdata is licenced under LGPL 2.1
- GData protocol is openly-documented, but not open-source
- Should we be encouraging use of non-open services?
- Opposition to inclusion of libgdata in GNOME's desktop release suite

There's some debate over whether we should be embracing services such as Google's, which is quite right. The GData protocol is openly-documented, but Google retain control of its direction and server-side software.

I proposed libgdata for GNOME's desktop release suite, and there was about a 50:50 split between support and opposition on openness grounds. I personally think libgdata is useful, but having some similar open-source services would be much better.



Links

- libgdata: `http://live.gnome.org/libgdata`
- libgcal: `http://code.google.com/p/libgcal/`
- GData protocol: `http://code.google.com/apis/gdata/overview.html`

Philip Withnall
philip@tecnocode.co.uk

Thanks for listening. Here are some links.