

GeoClue for Maemo

May 31, 2007

1 Overview

I've listed some possible sub-projects of my SoC project here. Starting from the most urgent and important (from my POV), the list looks like this:

- **Maemo packaging** – essential to get testers / developer-users
- **Wiki and doc update** – first impressions count
- **New (positioning) abstraction layer**
- **Civic positioning support** – human readable location info is, IMO, essential
- **Maemo UI** – important, but may require some previous subprojects to be done
- **Precision info for positioning backends**
- **privacy handling** – important, but also possibly a lot of work

All of these points have a short section below. I'd really like comments and suggestions on the subprojects and their order – I've been working on this on my own (no proper internet access for about a week) so I've probably become blind to my own mistakes and omissions....

2 Modifications to geoclue proper

These are ideas that have popped up during the last month or so. they're not necessarily maemo-related, just things that might make geoclue better. Like I said, I'd love comments on the ideas and suggestions on which ones I should work on and in which order.

2.1 New (positioning) abstraction layer

Motivation: Applications should not have to specify which backend they want to use. Applications should be able to say "*I want / Coordinates / City Name / Street Address / ... /*" and receive the information. This may not be a major issue right now with just a few backends, but it will be in the future.

Implementation: I guess this is just another dbus service that uses the actual backends. It needs a configuration file that contains the desirable backend order.

Question: Any suggestions on whether this is a good idea and how it should be done? I haven't really got a clear idea here...

Question: Is the backend order something a user will want to change?

2.2 Precision information for positioning backends

Motivation: Especially if the new abstraction layer (see 2.1) is implemented, the positioning backends should report some kind of precision estimate if asked. Otherwise applications will have no idea if the accuracy is 10m (GPS) or 10km (hostip). This would also be useful if additional privacy measures (which might lower positioning accuracy, see 2.4) are implemented.

For reference, here is the Google geocode API accuracy chart: <http://www.google.com/apis/maps/document>

Implementation: Add a dbus method to all positioning backends.

2.3 Civic Positioning support

Motivation: applications will want to know other things than just coordinates, such as City Name or Street Address.

Implementation: The supported civic location types can be picked from IETF rfc 4119. The actual results could be received in several ways, e.g. *position by GPS + reverse geocode with geonames.org* or *position by GPS + reverse geocode with local city name database* or *manual input*. I believe this would be a whole new backend type.

Question: is there earlier work on something like this?

2.4 Privacy handling

Motivation: Location privacy is something that people take seriously. The issue is naturally more important for mobile devices than normal computers or even laptops. There should be a way for the user to set her preferences and this should definitely be handled by the platform (geoclue), not by individual applications.

Implementation: This will be fairly tricky to get right. The UI has to be simple, but the issue itself is pretty complex. Something to remember: geoclue cannot really prevent an application from accessing e.g. `gpsd` directly, so the privacy settings should be seen as a way for the user to tell well-behaving applications her preferences on this issue. With this in mind, I believe a lot of the complexity can be hidden: the applications should tell geoclue what default permissions they expect.

IETF GEOPRIV working group has some geolocation privacy documents (see the links section), and this is my current idea based on those:

- Two configuration items
 - list of privacy groups with simple access rules (it seems GEOPRIV considers these “meta policies”):

 world: *rules...*
 colleagues: *rules...*
 family: *rules...*
 me: *rules...*
 - list of entities and the group they belong to (not sure if this needs to be viewable for the user at all)

 maemomapper: me
 browser/default: world
 browser/www.jaiku.com: family
 instantmessenger/default: world
 instantmessenger/bob@example: colleagues
- Applications add themselves to the list of entities (with a sane default group). The list of groups is created by geoclue.
- The rules for the groups can internally include the *condition/action/transformation*-triplets from the IETF specs, this would leave the door open for location-conditions and other fancy rules. However, I believe a single slider is enough for the user at least on maemo platform: This slider would indicate both the coordinate precision that will be presented to the applications in the group and the access they will have to different civic location types. Example values of both of those variables: [100km/Country] – [10km/Region] – [1km/City] – [Exact position/Full civic location].

2.5 Wiki and documentation update

Motivation: The wiki needs some love: It's currently a fairly good collection of miscellaneous information, but it lacks focus. In my opinion it also lacks a "*Why should I care*" -section for the application developers (something I've been promising to write).

Implementation: Divide page into two or more pages. Some possibilities:

- Main page – Intro, contact info, repository info
- Detailed Information – Whatever passes as general architectural docs
- Development page – links, ideas, et.c

3 Maemo specific development

3.1 User interface

The original document on maemo.org has some mockups and such, but I tried to keep this one short for now.

Position Change information

This is applicable to only some backends, mostly civic positioning . There should be an option to manually correct a wrong position (see *manual position*).

Implementation: Notification, e.g. "*New location: Helsinki*"

Settings applet

The privacy settings, possibly the preferred backend order and other things will require a Control Panel applet. The implementation is highly dependent on what geoclue features get implemented (see e.g. *Privacy Handling*)

Manual position widget

A user may want to specify a position manually. This manual positioning has potential to become a very complex piece of code that almost works... will have to be careful not to start something I can't finish.

Implementation: dialog with text input field for location input + a map widget for selecting location. The input field only makes sense after civic positioning is in use.

Alternative implementation: send the user to a webmap site that supports something like this and get the position information from there. This would free us from developing a very tricky piece of UI...

Question: Is there a service we can use for the alternative implementation? Maybe a Google maplet with a draggable marker?

Desktop applet

A small applet showing the current location as text. This would be a nice demo, but (again) requires civic location support to really be useful.

Statusbar applet (deprecated)

I do not think there's much use for this, but I'll include it in the list as it was here in the original document...

3.2 Packaging

Motivation: This is a must-have pretty much from the beginning of the project. The packages should be available from a common repository as early as possible, otherwise application developers will not be interested in using geoclue. Packaging work will naturally continue through the whole project.

Creating Debian/Ubuntu packages might not be a large job on the side.

4 Links

Privacy

1. Geolocation Policy: A Document Format for Expressing Privacy Preferences <http://www.ietf.org/rfc/rfc4745.txt>
2. Geolocation Policy: A Document Format for Expressing Privacy Preferences for Location Information <http://tools.ietf.org/html/draft-ietf-geopriv-policy>

Location Object

1. RFC 4119: A Presence-based GEOPRIV Location Object Format <http://www.ietf.org/rfc/rfc4119.txt>
2. GEOPRIV PIDF-LO Usage Clarification, Considerations and Recommendations <http://tools.ietf.org/html/draft-ietf-geopriv-revised-civic-lo>
3. Revised Civic Location Format for PIDF-LO <http://tools.ietf.org/html/draft-ietf-geopriv-revised-civic-lo>