Planet SoC

Home › Blogs › jeff's blog

Week 10 Updates: Lists and Dictionaries

Tue, 08/07/2007 - 05:34 — jeff

Last week I released msimprpl 0.13, incorporating the improvements detailed in my previous blog post. This version was built against Pidgin and Libpurple 2.0.2, but it should also work with 2.1.0 (but I haven't tested it). However, merging msimprpl into the main Pidgin tree isn't too far off. To plan for the merge, I propagated the latest im.pidgin.pidgin to im.pidgin.soc.2007.msimprpl. The plugin compiled and ran without problems in the newest Pidgin.

In the meantime, I fixed bug #2400, a crash on Windows when deleting buddies. Linux isn't affected because it appears to handle null strings in printf without crashing, instead substituting an innocuous "(null)" string. Windows crashes. I fixed the crash by being more careful about accidentally passing a null pointer.

In msimprpl's font size conversion routines, I used the round() function to choose the closest value when translating from font size in pixels to font size in points or HTML's relative sizes. However, on some systems (bug #2281), round() is not available. I fixed this by implementing my own round() if it is missing, allowing compilation if this function isn't provided by the standard libraries, since it was only standardized as part of C99.

I did a comprehensive comparison of MySpaceIM and Pidgin emoticons, lining up each with the corresponding icon. It turns out that Pidgin has a wealth of emoticons available for use by protocol plugins, but only a small subset are enabled by default. Which smileys are enabled for which protocols is specified by the theme, so I edited the theme file to include the symbols recognized by the official MySpaceIM client. I haven't tested this yet, but if it works it will save a lot of code in myspace.c. Unfortunately, emoticon handling is complicated by the fact that emoticons are not sent as-is, but as XML tags, such as: <i n='bigsmile' />. There are also a handful of emoticons supported by MySpace but not yet Pidgin.

I greatly improved message "postprocessing"--that is, the ability for functions to send protocol messages to a username, but have it translated automatically to userid--by allowing all message element types to be postprocessed. Previously, the userid could be substituted into a string. Now, I pack the element (readying it for transmission over the network), perform the substitution, and replace the element with the new element, as a raw string. This allows postprocessing to take place within list and dictionary items, or even more complicated data structures not yet imagined. As long as the data can be serialized to a string (which all elements can, because they eventually end up on the network), it is possible.

This improvement to postprocessing allows for more extensive use of appropriate data structures within msimprpl. I finally added a working dictionary type and use it in several places for sending messages. What I mean by the term "dictionary" is an ordered associative array, as often used within the "body" field. I initially used glibc's GHashTable for this purpose, but it does not satisfy these requirements:

- Ordered key/value pairs
- Possibly duplicate keys, with different values

A hash table is therefore not the best choice, and O(1) access to the elements is not that critical anyway.

My solution was to use another MsimMessage nested within a MsimMessage, to represent the dictionary type. MsimMessages satisfy all the requirements and make a perfect fit. Another benefit is that packing the dictionaries to a serialized format for network transmission can be done by reusing much of the same code that msim_msg_pack() uses.

That's about it for this week. In the coming weeks, if all goes well msimprpl will soon be integrated into Pidgin itself, bringing many more users along with it. Hopefully I'll get most of the major bugs worked out by then.

Tags: msimprpl Pidgin
Source: jeff's blog
Login or register to post comments

Comments

Big ups!

Tue, 08/07/2007 - 12:52 — Anonymous

Sounds awesome Jeff. Pidgin has been missing MySpace for FAR too long. It sounds like not only are you doing it, but you’re doing a very legitimate implementation. I for one can’t wait to use it!

Login or register to post comments