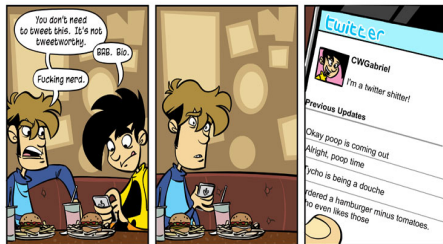# Making Twitter Suck Less With Perl

Speak Perl to your favorite μblogging service

What sucks, but ain't Twitter's fault?

- Following annoying people
- Having boring friends
- Twittershitters

What actually sucks?

- Interesting tweets buried in a sea of nonsense
- Tag spam (#omg #its #hashtags #allthewaydown)
- Skeezy webapps that steal your credentials
- Worms (StalkDaily)
- Marketing bots
- SEO crazies

Perl can help with most of these!

How can Perl help?

- Use the power of CPAN!
- Automate finding what you want
- Filter out what you don't care about

Net::Twitter

- Uses Moose
- Works with Twitter and Identi.ca
- REST, Search and Twittervision API support
- OAuth and Basic Authentication
- 1 to 1 mapping with API
- Plenty of syntax sugar
- Net::Twitter::Lite
  - Fewer Dependencies
  - Same great taste

Net::Twitter Basic Use

```perl
my $nt  = Net::Twitter->new({
            username => $user,
            password => $pass,
            traits   => [qw/API::REST/],
        });
my $my_tl    = $nt->user_timeline({count=>5});
my $buddy_tl = $nt->user_timeline({
                  screen_name => $buddy
              });
$nt->update('Perl rocks!');
$nt->new_direct_message($buddy,$tweet);
$nt->follow_new($cool_tweep);
$nt->unfollow($seo_guy);
```

WWW::ItsABot

itsabot.com has a constantly-updated database of Twitter bots.

```perl
use WWW::ItsABot qw/is_a_bot/;
my $username = 'foobar';
if ( is_a_bot($username) ) {
    print "$username is a bot\n";
} else {
    print "$username is not a bot\n";
}
```

It ain't perfect, but it is pretty good. It may think some celebrities are bots and, really, it isn't that far off.

Twitter::TagGrep

grep for Twitter timelines

```
my $tg = Twitter::TagGrep->new(
            prefix => '#!',
            tags => [ @tags ]
        );
my $timeline = $twit->friends_timeline({count=>200});
my @matches  = $tg->grep_tags($timeline);
for my $tweet (@matches) {
    print join(', ', @{$tweet->{tags}}), ": ",
              $tweet->{text},"\n";
}
```

Log::Dispatch::Twitter

Add a Twitter user as a Log::Dispatch logger easily!

```
use Log::Dispatch;
use Log::Dispatch::Twitter;
my $logger = Log::Dispatch->new;
$logger->add(Log::Dispatch::Twitter->new(
    username  => $username,
    password  => $password,
    min_level => "error",
    name      => "twitter",
));
```

Task::Twitter

Get the modules mentioned in this talk easily:

```
sudo cpan Task::Twitter
```

Hitting a CPAN mirror near you soon!

What I'm working on

- Visualizing Twitter connections with GraphViz
  - Colorized based on user characteristics
- Extendable Twitter Bot framework

What are you working on?

Conclusions

- Most of the annoying things about Twitter can be solved with a few lines of Perl.
- There is an amazing amount of data waiting to be tapped.
- Perl to the rescue!

Thanks!

- Open Source Bridge volunteers
- Larry
- Marc Mims for Net::Twitter
- The Twitter Dev Team
- All my fellow Open Source Citizens