Driving Google Apps with LDAP

Syncing your directory using GADS

Boyd Duffee - Keele University

GADS Google Apps Directory Sync

Automatic provisioning of

- user accounts and profiles
- groups
- resources
- contacts
- org units (unused at Keele)

GADS

- selects objects to provision using LDAP filters
- maps LDAP attributes to Google attributes
- fetches data from both sources, compares the lists and applies CRUD operations to Google
- test sync to observe potential changes
- a one-way sync - LDAP isn’t altered

GADS

- authenticates Google with OAuth, works through proxies
- can connect to SSL enabled LDAP servers
- writes activity to logfiles, sends notifications via email

I am not a number!

userids => uniqueIdentifiers => email

- cca13
- uniqueIdentifier=61822192-3E00-11DD-B2C8-8399CB0AF2BD,dc=people,dc=keele,dc=ac,dc=uk
- b.duffee@keele.ac.uk

- multiple domains tricky to manage in Google
- loss of domains meant emails changed

j.smith@domain.keele.ac.uk => j.smith2@keele.ac.uk

Google Groups

What are groups used for?

- maillists, web forum
- shared Calendar events
- sharing Google Docs
- controlling Resource visibility

Google Groups

May turn off the ability of your users to create their own groups (conflicting reports)

Want to know more?
http://learn.googleapps.com/products/groups

Group Management

Make life easy

- Automatic groups derived from data
  - HR for staff groups
  - student modules - trialing 330 groups
- Manual groups
  - devolve membership management
  - add members/owners to automatic groups

LDAP Schema

```
objectclass ( kdirobjectclass:9 NAME ‘kdirGoogleGroup’
    SUP top STRUCTURAL
    MUST ( displayName $ cn )
    MAY ( uniqueMember $ description $
          mail $
          owner $ manager $
          kdirAccountType $ kdirDeptCode ) )
```

LDAP Schema

```
objectclass ( kdirobjectclass:9 NAME ‘kdirGoogleGroup’
    SUP top STRUCTURAL
    MUST ( displayName $ cn )
    MAY ( uniqueMember $ description $
            mail $                      sometimes used for external emails
            owner $ manager $           who can modify the membership
            kdirAccountType $ kdirDeptCode ) ) for future use?
```

An aside

a short detour describing how I developed the tools to interact with our LDAP and my plans for future development …

Software development

Interrupted environment

- Test Driven Design, modular, self-contained code
- business logic separated from implementation
- refactor on the third copy

Business Logic

```
#!/usr/bin/perl
use Modern::Perl;
use Keele::Utilities::LDAP 0.03;
use Getopt::Std;
our($opt_g, $opt_v, );        getopts('g:v');

my $group = $opt_g if is_google_group($opt_g);
die "$opt_g is not a valid group" unless $group;

my @new_members = grep { is_valid_keele_userid($_) } @ARGV;
die "No valid keele userids in argument list" unless @new_members;

add_google_group_owner( $group, \@new_members)  or die "Couldn't add owners to $group";
add_google_group_member( $group, \@new_members)
      or warn "Couldn't add members ", join(', ', @new_members), " to $group";
```

When to refactor

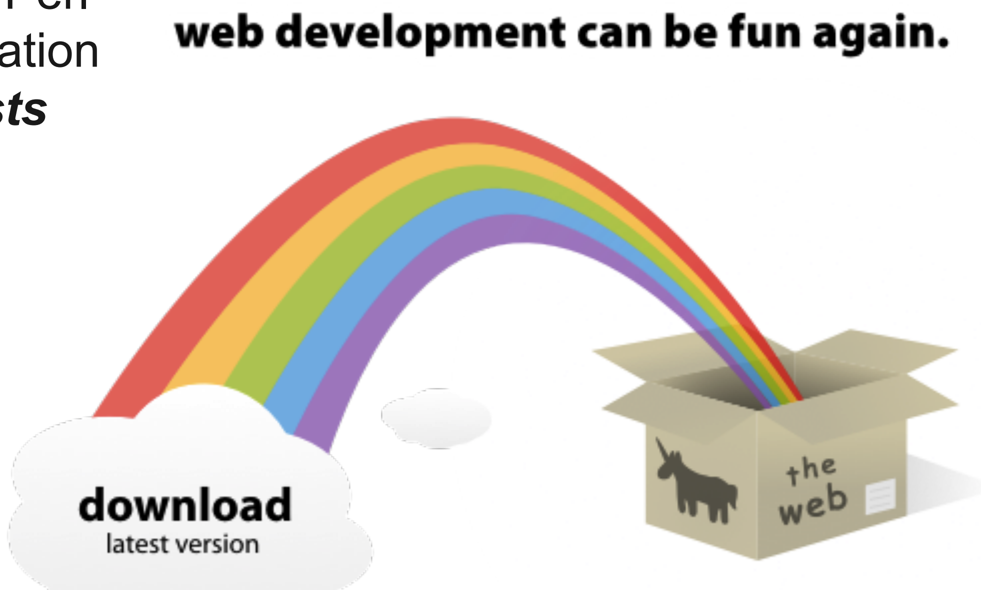
```
sub get_all_google_groups {
        my ($options) = @_;                                       # options in red, defaults in blue
        my %filter = (filter => $options->{filter} || "cn=*" );
        my $searchbase = exists $options->{base}     ? join ',', $options->{base}, $googlegroups_baseDN
                                                     : $googlegroups_baseDN;
        my $return = exists $options->{return} ? $options->{return} : 'cn';

        my $ldap = get_ldap_connection();
        my $mesg = $ldap->bind( $googlegroups_baseDN ) or croak "Couldn't bind to $googlegroups_baseDN: $!\n";
        my $searchobj = $ldap->search(  base   => $searchbase,
                                        %filter,
                                        attrs => [$return],
                           );
```

Model View Controller

Mojolicious

http://mojolicio.us/

- one of 3 major web frameworks in Perl
- MVC separates logic from presentation
- excellent video tutorials - ***Mojocasts***



Re-use library code developed for scripts

Website design

Required to devolve membership management

- authenticated sessions
- functionality through library calls
- screens present data to members
- allows owners to add/delete members

… back to LDAP and GADS

Calendar Resources

- used for booking rooms & equipment
- suggested naming schemes
  - city-building-floor-resource type (Boardroom)-id (room number)
- my LDAP tree
  - cn=laptop-1,dc=finance,dc=calendar,dc=apps, ...
- move the resource to new branch when a department leaves the building

Calendar Resources

- used for booking rooms & equipment
- suggested naming schemes
  - city-building-floor-resource type (Boardroom)-id (room number)
- my LDAP tree
  - cn=laptop-1,dc=finance,dc=calendar,dc=apps, ...
- move the resource to new branch when a department leaves the building

Contacts

2 conflicting interests

- auto-suggest allows you to find people as you type the name
- private mail aliases now advertised (oops!)

… it’s on the “to do” list

/dev/random

a few issues that I met along the way …

Unicode

1.3.6.1.4.1.1466.115.121.1.15 is UTF-8

convert Unicode to UTF-8 with perl

```
utf8::encode($string);  # "\x{100}"  becomes "\xc4\x80"
```

in-place operation

```
utf8::decode($string);  # "\xc4\x80" becomes "\x{100}"
```

Unicode::UTF8 is faster! more robust?

```
$octets = encode_utf8($string);
```

Unicode

Does your application need a Byte Order Mark?

```
print $fh chr(“\x{feff}”);    #or chr(0xFEFF)
# for UTF-8, the BOM is \xEF\xBB\xBF
```

at the beginning

Other uses for LDAP

Action Logging

- Registration suite records all account modification events in LDAP

Phone system

- Mitel had an integration with LDAP, now it seems to be only Active Directory

Thank you

Boyd Duffee

b.duffee@keele.ac.uk

Keele University

http://www.keele.ac.uk