Managing SBC People pages

Updated 2009

1. Overview:

one perl script drives the views. The script can accommodate multiple stages, which are subroutines, with one stage called as the default. Data are pulled from the LDAP, and formatted into views using the perl template toolkit module. A 'wrapper' is also added with the SBC web page boilerplate

2. Web Application Components:

2.1. cgi/
one perl script drives the views: ldapweb2009.cgi
The script can accommodate multiple stages as subroutines. I've learned better OO-perl while working with this script.

2.2 Stages and subroutines:
2.2.1 Return everyone in the sbc tree who has a "title" field filled in:

```
  'showAllByTitle'  => \&showAllByTitle,
```

2.2.2 Return the results of a search on the name, email address or description

```
  'searchform'      => \&handleSearchForm,
```

2.2.3 Return an individual's profile page

```
  'showindividual'  => \&showIndividual,
```

2.2.4 Return a list of people with NSF-roles. Not used- or could be made more generic to show groups of people.

```
  'showPIs'         => \&showPIs,
```

2.3. lib/
The templates for each view, and the names should be pretty clearly matched to their subroutines. The most recent versions have '2009' in their title, since the views were changed somewhat from earlier versions.
people_showAllByTitle2009.tmpl
people_searchResults2009.tmpl
people_showIndividual2009.tmpl
people_nsfPIs2009.tmpl

Note: the earlier versions (people_* with no year) are still available. Sometime soon, you should add subdirectories for different website areas using templates:that is, these templates will be moved to lib/people

3. LDAP

3.1 GQ: We use the unix utility ‘gq’ for managing content in LDAP. The script searches the tree o=sbc,o=lternet,o=edu. There are many people who need to show in the SBC pages, but are already included in the PISCO tree, (and vice versa) so they don’t need to show twice. However, PISCO has its own uses for LDAP fields, and has already created dummy accounts to use this way. SBC does something similar, but our dummy accounts have the string “dummy” attached, to make it clear what they are. These entries do not allow login – people need to be sent to their original account to reset passwords, etc (see below for the jsp to reset passwords)

People page content: we are using many standard fields from the LDAP tree. But like PISCO, we have hijacked some fields for our own use. A better solution would be to create some custom fields, but that is outside my (a) field of expertise and (b) time horizon. (I felt guilt, till I saw that PISCO had already done exactly this with their tree) So SBC made use of the same set of fields.

3.2 Here are the current mappings between the LDAP and the people templates:

LDAP FIELD	showAll/searchResults	showIndividual	CONTENT
cn	content, search	content	common name
title	only those in list are returned, note #1	no	role in sbc: controlled vocab, see note #1
description	content	staff people, yes	area of expertise, 2-3 words
mail	content	content	email address
businessCategory	content	no	see note #2
employeeNumber	key, see note #3		lternet id
labeledURI	no	content	homepage
postalAddress		content	mailing address
employeeType	no	content	job title at address
telephoneNumber	content	content	phone number
destinationIndicator	no	content	research interests, longer than description’
registeredAddress	no	content	personal statement
carLicense	no	students: degree pgm	string, MS, PhD, etc

3.2.1 LDAP content notes
Note #1
title field: this is the role in SBC. The script uses these to select from the LDAP tree. People with no title have no role, so don’t show. There are alumni categories also, which have no applications written to view them yet.

Titles we are using: **LeadPI, Investigator_L1, Investigator_L2, Coordinator, GS, OP**. The labels for each group are set in the template. So Investigator_L1 in 2009 means NSF-signing-PIs, but in the future, it might refer to some other group, like the site’s executive committee.

Alumni titles (to date): **alumni-student, alumni-staff, alumni-postdoc**. The terms were changed from ‘GS’, etc, because the script search uses wildcards (ie, *GS*), so it will find the alumni when it should not.
NOTE: what you should do: if the vocabulary is truly “controlled” then it would be better to make the strings exact – then you could use the wildcard to return all GSs, past and present with *GS*.

Note #2
businessCategory field: This field is currently used only to hold a second ‘description’ info bit for coordinators because one of them has 2 roles (Jenny is also an investigator, and needs to show up twice with different descriptions). It could also be used for other people with multiple roles, also. A better way might be to find a way to use additional description fields. Then you’d have to standardize which role went with which position.

Note #3
employeeNumber field: this field holds the person’s id in the network system. This is standardized to be first-initial-last-name (e.g., mobrien). Potential conflicts have a ‘1’ added (confirm). We use this to value to key into the LDAP and get the person’s hash for the individual profile page. Eventually, it could be used to export our LDAP content as import for the network DB.

That’s about it.

<u>Changing passwords:</u>
This application is for display only. A second application exists, which can be used for administration.
http://sbc.lternet.edu:8080/sbcPeople/directory.jsp
This app is a copy of an old app written by Jordan for pisco, which (since I don’t know Java) I have never finished for SBC, or converted it to the jsp tag library. But it has 2 uses:
1. the search is more generic, so everyone in LDAP shows, including alumni and those with no title.
2. individuals can log in and change their passwords, and it can be used by the IM to encrypt passwords when creating accounts. (more on that in the IM guide How-to section.).
3. If you wanted to keep this for an admin interface, you could finish this up to some level
3.1. take some of the editing options off – for example, people should not be allowed to change their roles.
3.2 The page allows uploading of photos – take advantage of this for the profile pages in the production app.