

We Tagged Everything

- Concept of Tagging easy to understand for users
- Tagged Plone Content-types: News, Files, Conference Events and Pages
- Tagged Attendees: User Profiles (using Plone membrane product)
- Tags meant to classify and pull together content (e.g. identity+management, Authentication, idmashup06), and grow dynamically
- All unique Tags show up in left portlet - filtered by Our Tags and My Tags

Using the Tags: Member Profiles

- *membrane* is a Plone 2.5 way of customizing Members of the site. An Archetype schema is used to define the profile
- Each user has their own profile - and can pick an existing tag or add their own to signify their interests

Using the Tags: The Taggregator

Each tag has its own page, which is a mashup of various Plone content and external Blogs

Brief Demo of IDMashup.org

Let's walk through Home page (Blogs), Tags (e.g. identity+management, Authentication), Tag Portlet, Taggregator

identitymashup community hub

HOME + ABOUT + SCHEDULE + LOCATIONS + WIKI + SEARCH

LOG IN JOIN

OUR EVENTS

COMPLETE SCHEDULE

OUR TAGS

3D+Models
Accountability
Acoustic
AJAX
anonymity
augmented+demo...
Authentication
Avatars
Bandit
broadcast
business
Caribbean
Characterization

OUR NEWS

O'Reilly Network publishes article on ID Mashup
Andy Oram wrote an extensive article called "The Long View of Identity" on open identity and the ID Mashup

Welcome

This is the Community Hub for participants of the Identity Mashup Conference. Learn about this site and the greater conference environment.

The conference is over and we are working to transform this site into a resource with lasting community value. Stay tuned.

identify	connect	participate
Join this site, create your profile and see it in action. Experiment with other id apps.	Find people and content of interest. Interact with others in Second Life.	Participate by contributing to the IDMashup Wiki.

To help orient you, we have prepared a guide to this site.

Conference News

O'Reilly Network publishes article on ID Mashup
Andy Oram wrote an extensive article called

Featured Identity App

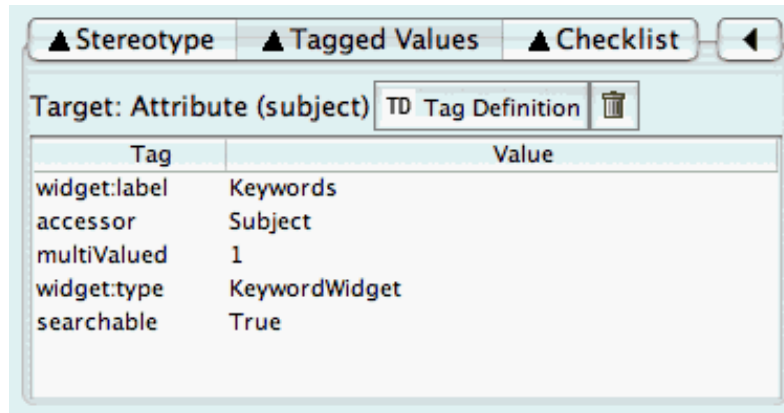
Heard about Higgins? It is an open source framework that integrates identity and

How do we Tag our content in Plone?

- Decided to use Plone's built in Keywording
- Easy to add Tags to your own Archetype-based content-types via ArchGenXML or editing your schema by-hand

Adding Tags via ArchGenXML

Or if you're generating your content via ArchGenXML, add a field called "subject" of the "lines" type, and make sure you use these tagged values:



Adding Tags to your Schema by hand

Adding the Keywords field to your own new Archetype content-types by adding a "subject" field like this:

```
LinesField(  
    name='subject',  
    widget=KeywordWidget(  
        label="Keywords",  
        label_msgid='mycontent_label_subject',  
        i18n_domain=' mycontent ',  
    ),  
    accessor="Subject",  
    searchable=True,  
    multiValued=1  
)
```

Unhiding the Keywords Field

For some content-types, Keywords should be front-and-center and not out-of-sight in the Properties tab. With Archetypes, it's easy to move this somewhere else - like on the primary edit form. (We subclassed many of the default content-types just to move this field)

```
##code-section after-schema #fill in your manual code here
```

```
# move subject field to main form, and put it under 'text' field  
ConferenceNewsItem_schema['subject'].schemata='default'
```

```
ConferenceNewsItem_schema.moveField('subject', after='text')
```

```
##/code-section after-schema
```

The Taggregator

Each tag has its own page, which is a mashup of various Plone content and external Blogs (for only 17 top tags). Three examples:

```
<div class="boxContainer"
    tal:define="results python:here.portal_catalog.searchResults(
        portal_type = ['ConferenceFile', 'ConferenceImage',
'ConferenceLink',
        'ConferenceNews ConferenceItem', 'News Item'],
        Subject =here.REQUEST.tag
    );">...display content titles here...</div>

<tal:attendees define="results python:context.portal_catalog.searchResults(
portal_type = 'SharedProfile', Subject =here.REQUEST.tag )">...display attendee names here...</tal:
attendees>

<tr tal:define="posts python:here.parseTechnoratiFeedForTag(here.REQUEST.tag)" tal:condition="posts">
...display blog titles here...</tr>
```

Pulling in Blogs

- Our audience had their own blogs; let's not reinvent in Plone
- Decided to leverage Web 2.0 blog sources
- Researched most relevant 17 tags for this conference

idmashup06, digital+identification, identity+eclipse, identity+management, foaf, openid, yadis, digital+id, identity+higgins, sxip, socialphysics, identity+lid, identity+authentication, identity2.0, saml, sso, digital+identity, planetnetwork

- (In the Taggregator pages, we wanted to pull in blogs for ALL tags, but too many Technorati queries)

What is Technorati?

- A search engine for blogs with an API
- Imagine trying to support all blog formats?
- <http://www.technorati.com/about/>

Technorati queries in Plone

1. An external Python script that connects with Technorati via their API (Extensions/technorati.py) plus a minor External Method to connect with script (technoratiGetBlogsByTag)
2. A minor Python Script used to Cache results (getTechnoratiFeed.py)
17 Technorati queries (one for each tag) once per hour = 400 requests limit per day
3. A page template that shows the results on the home page

Technorati: Main Python script

-
- Need a Technorati License Key
 - We began with a Python wrapper for the Technorati API http://www.myelin.co.nz/technorati_py/
-

Technorati: Main Python script (slide 2)

We walk through each tag and query Technorati

```
def getBlogsByTag():
    blogList = WC_DEFAULT_TAG_LIST
    limit = 0

    # Functional code ...
    masterResultList = []
    for tag in blogList:
        _log("Processing tag '%s'" % tag)
        queryResults = tagQuery(tag)
        _log("Found %d items for specified query" % len(queryResults))
        masterResultList += queryResults
    sortedList = _sortListByDateTime(masterResultList)
    setSize = len(sortedList)
    if limit > 0:
        sortedList = sortedList[setSize-limit-1:setSize-1]
    sortedList.reverse()
    return sortedList
```

Technorati: Main Python script (slide 3)

We then call this function for each tag

```
def tagQuery(query):
    # Toolkit level interface function. Performs some cooking of the raw output of
    # '_apiTagQuery'.
    _log("Querying for '%s' ..." % query)
    rawResults = _apiTagQuery(query)
    cookedResults = []
    if rawResults is None:
        _log("No results found for this query.")
    else:
        _log("Results found. Cooking ...")
        tempResults = rawResults[WC_TAG_QUERY_SERVICE]
        for oldItem in tempResults:
            newItem = oldItem['weblog']
            newItem['created'] = oldItem['created']
            newItem['postupdate'] = oldItem['postupdate']
            newItem['tag'] = query
            newItem['title'] = oldItem['title']
            newItem['excerpt'] = oldItem['excerpt']
            cookedResults.append(newItem)
    return cookedResults
```

Technorati: Connect with Plone and Caching

This is the script we Cache once per hour for performance and to stay within the Technorati query limit

```

## Script (Python) "getTechnoratiFeed"
##parameters=
##title=Wraps technorati function so we can cache it

try:
    return context.technoratiGetBlogsByTag()
except:
    return []

```

Technorati: *template_hotBlogs.pt*

```

<tal:hotBlogs metal:define-macro="hotBlogs"
    tal:define="theposts python:here.getTechnoratiFeed()[:5]" >

<h3>Fresh Blogs!</h3>
<h4 class="subtitle">ID Mashup in the blogosphere - tag with idmashup06</h4>
<tal:check tal:condition="theposts">
    <ul class="no-indent mainText">
        <tal:repeat-block tal:repeat="post theposts">
            <li class="no-bullet blogTitle"
                tal:define = "postDate python:DateTime(post['created']).strftime('%b %d %l:%M%p');
                    tag python: post['tag'];"><a href="
                tal:content = "python: post['title']" tal:attributes="href python: post['url'];
                    title python: post['excerpt'];">

                Blog Post Title
            </a><br />
            <span style="font-weight:normal;font-size:80%;">
                <span tal:content="python: post['name']">Opinity Blog</span>
                on <a href="
                    tal:attributes="href string:${portal_url}/taggregator?tag=${tag}"
                    tal:content="python:tag.upper().replace('+',' ')"
                    style="font-size:90%;">Yadis</a>,
                <span tal:content="postDate">6/15/06 3:15PM</span>
            </span>
        </li>
    </tal:repeat-block>
</ul>
</tal:check>
</tal:hotBlogs>

```

Bringing in Tagged Photos via Flickr

Pulling in images from Flickr is very easy. Here's how to grab all the images tagged with 'idmashup06':

```

<!-- Start of Flickr Badge -->

<script type="text/javascript"
src="http://www.flickr.com/badge_code_v2.gne
?count=5&display=random&size=t&layout=h&source=all_tag&tag=idmashup06"></script>

<!-- End of Flickr Badge -->

```

Some Tagging Issues

- Tag pollution

- Tags with different meanings (i.e. Tags in different contexts, or different in Plone vs Internet) - more specific is better or boolean functionality
 - Tags aren't necessarily descriptive
-

Other Site Features

- Conference attendees saw a customized schedule based on their profile
 - Site supported a specific conference, but serves the community after the conference has ended: Top Blogs, Plone content, User Profiles.
 - Sync'ing Plone 'membrane' Profile with Higgins Digital Identity Framework
Higgins is an open source framework that integrates identity and relationship information across systems.
 - OpenID available for Plone (not used here)
-

In Closing: Mashup Ideas to Think About

- Plone is a good platform to make your data available to others, as well as "consume" data from a variety of sources. Thanks Python!
 - There is a growing list of Web 2.0 Services making content available - Google Maps, Technorati, Yahoo Geocoder, eBay API, Basecamp API, Amazon API, etc - there are now geocoding and Google Map Plone products now.
 - AJAX is now a core feature of many Mashups (e.g. Google Maps) so worth learning
 - Consider making your data available to be "mashed up" on other sites by making it available in easily digestible formats - not just HTMLRSS is one example format - GeoRSS, XML - or provide your own API.
-

Thank you

- Questions from the Audience
- Presented by

Brian Gershon of RagingWeb.com

briang@ragingweb.com * (206) 347-0916 * Seattle, WA

- idmashup.org sponsored by [The Berkman Center for Internet and Society at Harvard Law School](#), [The Interra Project](#), [MIT Media Lab](#)
- Thanks to the whole development team (alphabetically): Brian Gershon, Christopher Gronbeck, Derek Hoshiko, Eric Magnuson, Jon Ramer, Alex Tokar, and Bryan Wilson.