

WTF is Zope?

- History
- Zope 3
 - Differences between 2 and 3
 - Repoze.*
 - Zope 3 re-imagined
 - Zopeproject
 - repoze.bfg
 - Grok



Python Web Frameworks

- When I think of python web frameworks...
 - Django
 - Turbogears
 - Pylons
 - Flavor of the month....
- And then there's Zope. This big huge mystery that's been around forever quietly doing it's own thing.



History of Zope

- In the beginning there was Bobo.
 - Generally regarded as the first python web framework
 - 2 core ideas
 - Map urls to objects
 - Call objects to generate an HTTP response
 - Written in 1997
 - A time when there were kick ass technologies like....
 - Classic ASP
 - CGI
 - JSP
 - PHP



History of Zope(cont'd)

- Then there was BoboPOS
 - Acid Complaint Object Oriented Database
- DocumentTemplate
- Zope2 = Principia(App Server) + Bobo + BoboPOS + DocumentTemplate



History of Zope(cont'd)

- On top of Zope2 other features were built as needed like....
 - Container based security model
 - "Through the Web" development. Create your application right in the web browser admin interface



History of Zope(cont'd)

- Guido was working there
- Zope was released as open source(1998)
- A vibrant community flourished
- Plone, a zope based content management system came to be
 - A vibrant community flourished



History of Zope(cont'd)

- meanwhile...
 - J2EE (1999)
 - Java: Struts (2001)
 - ASP.net (2001)
 - Work on Zope 3 begins (2001)
 - First release of Zope 3 (2004)
 - Ruby on Rails(2005)
 - Turbogears (2005)
 - Django(2005)
 - Pylons(2005)



History of Zope(cont'd)

- Ahead of it's time?
- Influential?
- Stuff worth using? Stealing? Learning?



Complaints about Zope 2

- Steep learning curve
- Very heavy
 - in order to play in zope 2 your objects had to inherit from a lot of base classes/mixins and special attributes.
- Tightly coupled



Zope 3

- Written from the ground up to address the complaints of Zope 2
- Agile test driven development process
- Emphasis on documentation
- Less emphasis on "Through the Web" development.
- Smaller independent parts



Zope 3 (cont'd)

- A collection of free software
 - Jointly developed by the Zope corporation and a very large community
 - Can be used in whole or in part
 - Manage complexity in gluing software components together
 - Securely publish objects on the web or other systems
 - Make it easy to do quality assurance



Differences between 2 and 3

- 2 was intended to be a content management system that just happened to be powerful enough to build any web application on.
- 3 intended to be component based framework to build applications(even for the web), you could even build a content management system(even for the web).



Differences between 2 and 3 (cont'd)

- In 2 you extend by inheriting from some base object, and monkeypatch
 - smells like asp.net
- In 3 , component based architecture configurable through an xml dialect called zcml
 - smells like spring



WTF is repoze.*?

- Integrates zope technologies with WSGI and reusable middleware. Like...
 - <http://repoze.org>
- `repoze.tm2`: middleware that implements a transaction policy.
- `repoze.who`, `repoze.what`: authentication/authorization framework. Used by `turbogears 2`



WTF is repoze.*?(cont'd)

- `repoze.tempita:filter` which conditionally causes the body returned by the application to be run through the Tempita templating engine
- `repoze.catalog`: indexing/searching engine similar to lucene. 100% python.



WTF is repoze.*?(cont'd)

- `repoze.formapi`: small framework to take you through the entire process of rendering a form, provide default values, validate and execute form actions. Alternative to `formencode/htmlfill`
- And many more!!!



Zope 3 re-imagined

- Now that zope is more a collection of technologies, several flavors of zope as a full-stack have come about
 - Zopeproject
 - repoze.bfg
 - Grok
 - Probably others



Zopeproject

- Egg installable full stack zope 3 with all the bells and whistles.
- Python 2.5 only
- Recommended for those who are familiar with zope, and want to use it.(Like zope 2)



repoze.bfg

- Egg installable minimal stack zope 3
- Works on 2.6
- Recommended for those who are familiar with wsgi web applications, but not zope and don't know how much zope they might want/need.(like pylons)



Grok

- Egg installable rapid development framework based on zope 3
- Works on 2.6
- Recommended for those who prefer convention over configuration and might not care about zope one way or another.(like rails)



Code time?

