

Introducing CMF 2.0

presented to the
New Orleans Plone Symposium 2006
2006/03/09

Tres Seaver
Palladion Software
tseaver@palladion.com

Overview

- Where have we been
- What is in CMF 2.0?
- Where are we going?

Paris Sprint, March 2005

- Originally aimed at implementation of Zope3 ECM features
 - semantic / RDF indexing
 - skins framework
- Added focus on integration of Five into Zope2
 - Silva/CPS/CMF/Plone all run on Zope 2.8.x!

Importance of Five

- May 2005: Five 1.0.1 ships, along with Zope3 3.0, as part of Zope 2.8.0
- Five allows developers to begin using Zope3, while preserving Zope2 investment
- Five expands audience for Zope3 *dramatically*.

Goldegg Project, Fall 2005

- CIGNEX, with customer, wanted to invest in the Plone “stack”
 - Zope
 - CMF
 - Plone
- Initiative directed six-figure funding to prime collaboration

Goldegg Goals

- Advance the Plone product by investing in the “stack”
 - Make Plone / CMF / Zope “faster, stronger, smarter”
- Leverage the value of Zope 3 in the current stack
- Get the Plone community and the CMF community working more closely
 - Involve project leads / developers from across Zope ecosystem
- Improve Plone's process of making framework decisions
 - Future framework activities are done in the CMF, if possible
 - Have Plone focus on user experience

Non-Goals for Goldegg

- Porting Plone to Zope3
- Adding sexy new end-user features
- Inventing new architectures
 - Focus instead on consolidation / improvement

Goldegg-Supported Projects

- Pluggable content export / import
- Leverage Zope3 event model
- Content views as Zope3 view classes
- Improve packaging using Python “eggs”
- Improve workflow modeling in AGX

Goldegg Outcomes

- Shared sprints
 - Castle Sprint, September 2005
 - “Red zone sprint”, October 2005
 - Packaging sprint, November 2006
- CMF 1.6: “Stepping stone” using new Generic Setup
- CMF 2.0 beta
- Plone 2.5 preview
- Cross-pollination of Plone, CMF, CPS developer communities

New In CMF 2.0

- Requires Zope 2.9+, Five 1.1+
- Zope3 technologies
 - Implements Zope3 views for content
 - Publishes Zope3 events
 - Adapters for pluggability
- Architecture, examples for export / import of content
- Profile-driven site setup

Using Zope3 Views for Presentation

- Improves clarity
 - remove gnarly Python expressions in ZPT
 - Templates are managed / bound to the “view logic” currently found in scripts
- Improves maintainability
 - Views can be unit tested!
- Improves performance
 - Presentation logic can be trusted code

Configuring Views

- Filesystem software registers views via ZCML

```
<browser:page
  for=".interfaces.IIssue"
  name="index.html"
  class=".browser.IssueView"
  template="issue_view.pt"
  permission="zope2.View"
  layer="cmf" />
```

- View binds a template to a view class
- Views are registered for content interfaces

Anatomy of a View Class

- “Simple” Python code:

```
from Products.Five.browser \
    import BrowserView
from Products.L Leatherstocking.interfaces \
    import ITextTranscript
class IssueView(BrowserView):
    def getViewInfo(self):
        adapter = \
            ITextTranscript(self.context)
        url = self.context.absolute_url()
        info = {'transcript':
                transcripiter.render(url),
                'url': url,
                ....
                }
        return info
```

- Testable!

Zope3 View Templates

- “tal:define delenda est!”
- Not quite, but much reduced:

```
<div tal:define="info view/getViewInfo">  
<pre tal:content="info/transcript">  
TEXT TRANSCRIPT HERE  
</pre>  
</div>
```
- Template plugs values delivered by view into HTML

Issues with Zope3 Views

- Issue: customization?
- CMF 2.0 offers views as alternative
 - Skins remain default

Using Zope3 Events

- Why events?
 - Simplify code
 - Break dependencies
 - catalog reindexing
 - workflow notification
 - Allow pluggability

Event Types

- “Content” events (creation, modification)
- “Container” events (adding, renaming / moving, deletion)
- “Workflow” events (at transitions)
- “Timing” events (to drive scheduled work)

Publishing Events

- Who publishes?
 - View methods
 - Workflows
 - (Maybe) content methods
- Florent Guillaume landed Z3 container events in Zope 2.9
- Published where?
 - To the “global” Z3 event “service”, e.g.:

```
from zope.event import notify
notify(ContentAddedEvent(obj))
```

Responding to Z3 Events

- Subscribers registered by object type, event type
- Subscriptions replace legacy notifications in CMF 2.1
 - Catalog
 - Deferred indexing
 - Workflow tool
 - Automatic transitions
- Other possibilities
 - Logging
 - Content subscription
 - Cache invalidation

Why Content Export / Import?

- Revision control
- Staging
- Preserving content across software upgrades
- Interoperability (e.g., JSR 170)
 - Removes “vendor lock-in” on content

Content Export / Import Architecture

- Adapter-based serialization
- Variety of targets:
 - Zipfile
 - Filesystem
 - Parallel folder structure
- Exported content can be part of the CMFSetup “profile”
- Export / import views directly on content

Issues for Content Export / Import

- Need to revisit Z3's fssync
 - fssync is “lossless”; GS is “lossy” for low-level details
- Implementations for Plone content types
- “Partial” export
 - Export only one subfolder and contents

Profile-Driven Site Setup

- Profiles target site “types”
 - CMF's ISiteRoot, Plone's IPloneSite, even IPluggableAuthService!

Why Profiles?

- Similar use-cases to QuickInstaller's
- Profile setup expressed as data, not code
- Configured setup exportable
 - Snapshots
 - New filesystem profiles

Profile Types

- Base profiles initialize a site
 - ZMI add-list factory allows selection among profiles
- Extension profiles “add on” to a site
 - Tools
 - Skins
 - Workflows
 - Content Types
 - Content!

Endonde Vamos?

- CMF 2.0 Release
 - Another beta, by end of March
 - New profile selecting views instead of skins
 - Final release mid-April
- CMF 2.1 Goals
 - View customization
 - Improved event integration
 - Content class refactoring
- CMF 2.1 Release
 - Beta in late June
 - Release in July

View Customization

- Skin method customization is *valuable*
- Zope3 local component support clunky
- PyCon 2006, Philipp von Weitershausen prototypes for Five 1.4 / Zope 2.10

Customization Use Case

- Mark a folder as a “site”
- Inspect views for a content object, select one for customization
- Zope creates a custom PageTemplate derivative in the nearest site, configures as a view

Customization Issues

- We can customize the template; what about the view class?
 - Zope3 has incomplete support for persistent modules
- Alternative: allow embedding Python in a block in the template

```
<script type="text/server-python">
title = request.form.get('title')
def filteredValues():
    for x in self.contentValues():
        if x.someCustomMethod():
            yield x
</script>
<h1 tal:content="title">TITLE</h1>
<p tal:repeat="obj filteredValues"
    tal:content="obj/absolute_url" />
```

Event Integration

- Catalog as event subscriber
 - Implemented on a branch
 - Deferred indexing
- Workflow tool as event subscriber
 - Automating transitions after editing
- Workflow tool as event publisher

Refactoring Content Classes

- Poster child: CMFDefault.Document has too many responsibilities
 - Storage of “document source”
 - “Cooking” HTML at output
 - “Scrubbing” HTML at input
 - All-or-nothing on all this logic
- Refactoring Document
 - Adapters for cooking, scrubbing allow pluggability:
 - add new source types
 - change the “tag blacklist”

Zope2 / Zope3 Futures

- Original Vision
 - Zope3 evolves to fill Zope2's niche
 - Zope2 “withers away”
- Alternative
 - Zope3 continues as a technology platform
 - Zope2 re-absorbs most useful bits, purging cruft
- “Let's Make a Deal”
 - Zope2 community, and many Zope3 developers, favor “Door Number 2”

Lagniappe: Frostbite

- portal_skinfreezer tool
- Select skin path, then templates / scripts
- “Freeze” selected objects into a filesystem module
- Add a “freezer” to your skins tool, “shadowing” multiple paths
- Demo

Frostbite Status

- Frostbite 0.4 available
- Issues: code needs tweaking after generation

Q&A

Other Resources

- CMF 2.0 Roadmap Documents
 - <http://www.zope.org/Products/CMF/rqmts/>
- Goldegg Stack Project
 - <http://www.goldeggstack.org/>
- Frostbite
 - <http://palladion.com/home/tseaver/software/Frostbite/>

Thank you!

- Tres Seaver, tseaver@palladion.com