

CS 683 Emerging Technologies  
Fall Semester, 2006  
Doc 12 Some Web Issues  
Oct 3, 2006

Copyright ©, All rights reserved. 2006 SDSU & Roger Whitney, 5500 Campanile Drive, San Diego, CA 92182-7700 USA. OpenContent (<http://www.opencontent.org/opl.shtml>) license defines the copyright on this document.

## References

Patterns of Enterprise Application Architecture, Martin Folwer, 2003, Pearson Education

Seaside Source code, See <http://seaside.st/> for more information on Seaside

# Dynamic Pages

Embed Code in html

Generate html in code

# Embed Code in html

php  
Java Server Pages  
ASP

```
<HTML>
```

```
<BODY>
```

```
  This is a php example<BR>
```

```
  <?php
```

```
    $a = 1;
```

```
    $b = 2;
```

```
    echo "$a plus $b is $a + $b <BR>";
```

```
  ?>
```

```
</BODY>
```

```
</HTML>
```

# Benefits

Fast initial development  
Low development overhead

# Issues with Code in Html

Cant use code tools

- Syntax aware editors

- Debuggers

- Compilers

Web server is poor development environment

Difficult to manage large amounts of code

Code/logic repeated in multiple pages

Does not scale well

# Generate html in code

## Servlets

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class HelloWWW extends HttpServlet {
    public void doGet(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("
            "<HTML>\n" +
            "<BODY>\n" +
            "<H1>Hello World</H1>\n" +
            "</BODY></HTML>");
    }
}
```

# Issues with Generating Html in Code

Generating modern html is painful in code

Can't use HTML editing tools

# Model-View-Controller

As Web apps grew in complexity problems in development grew

People turned to MVC architecture

# Define MVC

## Model

- Domain logic
- Business policies
- Database interactions

## View

- Display of model in UI

## MVC Controller

- Input controller
- Forwards request to proper model

## Application Controller

- Flow of application
- Layer between view and model

# Motivation

Nonvisual entities are easier to test

Model and view

- Different things

- Require different skills

- Use different libraries

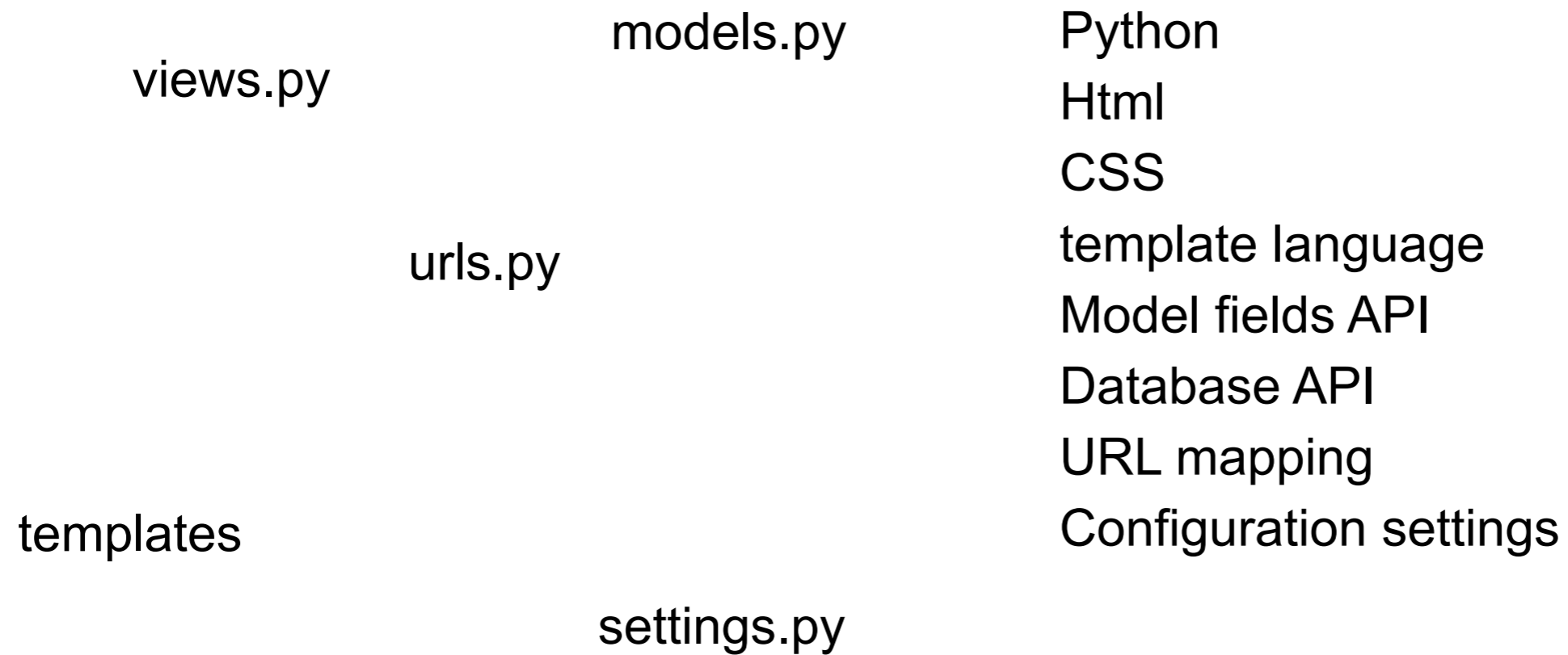
- Different rate of change

A model may need multiple views

# MVC & Web Frameworks

MVC is influencing Web framework architecture

# Django Parts



# Count Example

# Count with ++ and --

5

++ =

# urls.py

```
urlpatterns = patterns("",  
    (r'^count/$', 'cs683.adder.views.count'),  
    (r'^count/(?P<operation>.*)/$', 'cs683.adder.views.count'),  
)
```

# cs683/adder/views.py

```
from django.shortcuts import get_object_or_404, render_to_response
```

```
class Counter:
```

```
    def __init__(self):  
        self.count = 0
```

```
    def increase(self):  
        self.count += 1
```

```
    def decrease(self):  
        self.count -= 1
```

```
def count(request, operation=None):
```

```
    counter = request.session.get('count', Counter())
```

```
    if operation == 'plus':
```

```
        counter.increase()
```

```
    if operation == 'minus':
```

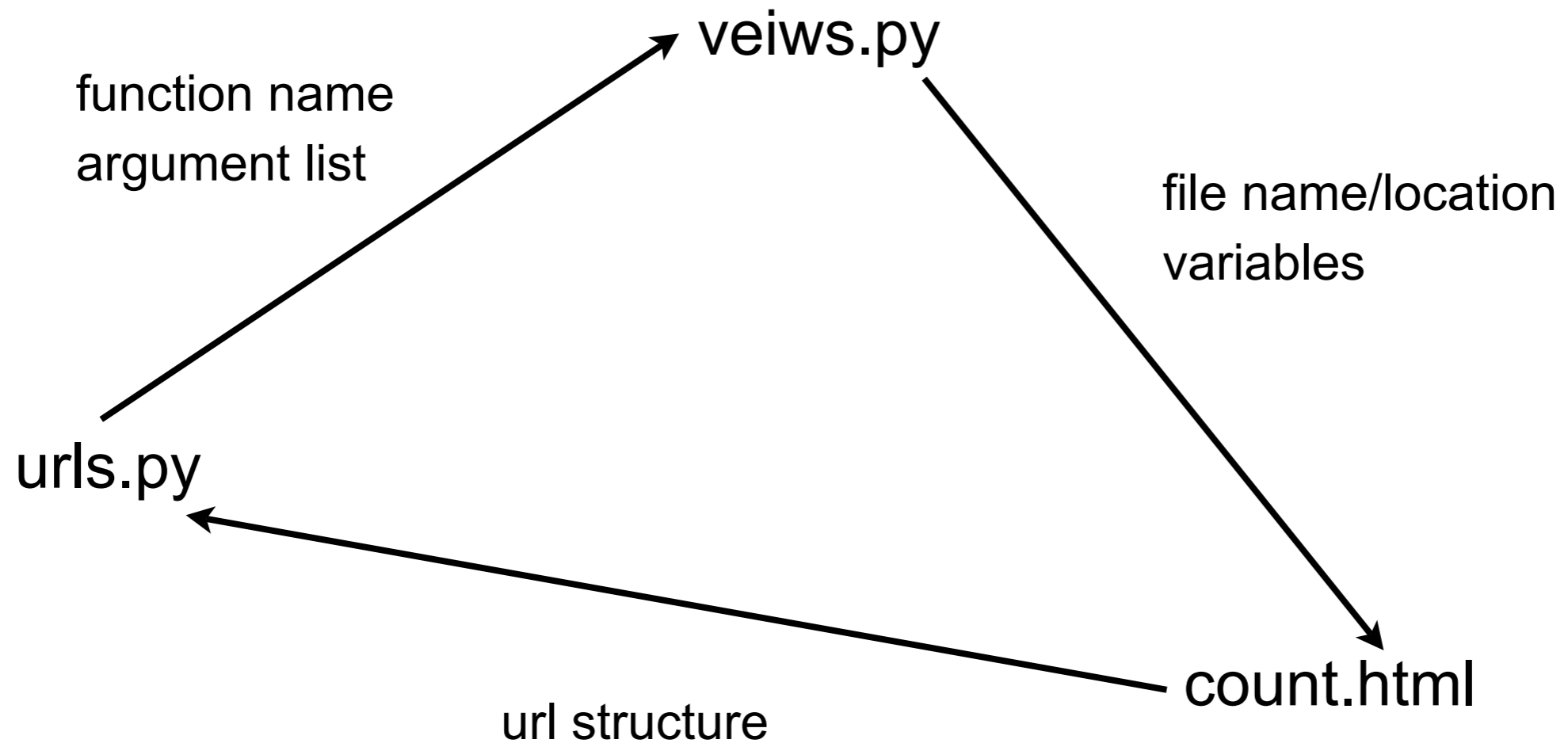
```
        counter.decrease()
```

```
    request.session['count'] = counter
```

```
    return render_to_response('adder/count.html', {'counter': counter})
```

# cs683/templates/adder/count.html

```
<h1>{{counter.count}}</h1>  
<a href="/count/plus">++</a>  
<a href="/count/minus">--</a>
```



# Back Button Woes

# What should happen?

Click on ++ five times

Click the back button

What should you see?

What do you see?

Now click on --

What should you see?

Seaside  
Generating html in code  
New approach

# Seaside

Continuation based Web application framework

Written in Smalltalk

Examples translated into Python

Interested in concepts

# Simple Example

```
class SimpleHtml(WAComponent)

    def canBeRoot():
        return True

    def initialize():
        registerAsApplication('simpleHtml')

    def renderContentOn(self, html):
        html.heading('Hello World')
        html.hr()
        html.paragraph('This is a paragraph')
        html.unorderedList(html.listItem(each) for each in ['cat', 'dog',
'rat'])
```

# Actual Smalltalk Example

```
Smalltalk defineClass: #SimpleHtml  
  superclass: #{Seaside.WAComponent}
```

```
!Class methodsFor: 'testing'!
```

```
canBeRoot  
  ^ true
```

```
initialize  
  (self registerAsApplication: 'simpleHtml')
```

```
!Instance methods!
```

```
renderContentOn: html  
  html  
    heading: 'Hello World';  
    hr;  
    paragraph: 'This is a paragraph';  
    unorderedList: [#{'cat' 'dog' 'rat'} do: [:each | html listItem:  
each]]
```

# http://127.0.0.1:8008/simpleHtml

```
<h1>Hello World</h1>
<hr />
<p>This is a paragraph</p>
<ul>
  <li>cat</li>
  <li>dog</li>
  <li>rat</li>
</ul>
```

## Hello World

---

This is a paragraph

- cat
- dog
- rat

[New Session](#) [Configure](#) [Toggle Halos](#) [Profile](#) [Terminate](#) [XHTML](#) 0/0 ms

# Counter Example

```
class Counter(WAComponent)
    def initialize():
        registerAsApplication('counter')
    def canBeRoot():
        return True
    def decrease(self):
        self.count = self.count - 1
    def increase(self):
        self.count = self.count + 1
    def renderContentOn(self, html):
        html.heading(count)
        html.anchor( callback =self.increase, text = '++')
        html.space()
        html.anchor( callback =self. decrease, text = '--')
    def __init__(self):
        WAComponent.__init__(self)
        self.session().registerObjectForBacktracking(self)
        count = 0
```

# Smalltalk Version

```
Smalltalk.Seaside defineClass: #WACounter  
  superclass: #{Seaside.WACComponent}  
  instanceVariableNames: 'count '
```

## Class Methods

```
initialize  
  (self registerAsApplication: 'counter')  
  
canBeRoot  
  ^ true
```

## Instance Methods

```
count  
  ^ count  
  
decrease  
  count := count - 1  
  
increase  
  count := count + 1  
  
renderContentOn: html  
  html heading: count.  
  html anchor callback: [self increase]; text: '++'.  
  html space.  
  html anchor callback: [self decrease]; text: '--'  
  
initialize  
  super initialize.  
  self session registerObjectForBacktracking: self.  
  count := 0
```

# http://127.0.0.1:8008/counter

0

[++](#) [--](#)

[New Session](#) [Configure](#) [Toggle](#)  
[Halos Profile](#) [Terminate](#) [XHTML](#) 1/0 ms

```
<h1>0</h1>
```

```
<a href="?_s=QPXeikFohLaHmopG&_k=pKjvdidM&1">++</a>&nbsp;
```

```
<a href="?_s=QPXeikFohLaHmopG&_k=pKjvdidM&2">--</a>
```

Programmer indicates callback method for link

Seaside generates html, urls to make it happen

The back button works correctly

# Components and Code Reuse

# Reusing Counter

```
class MultiCounter(WAComponent)

    def children(self):
        return counters

    def initialize():
        registerAsApplication('examples/multi')

    def renderContentOn(self, html):
        for a in counters:
            html.render(a)
            html.horizontalRule()

    def canBeRoot():
        return True

def __init__(self):
    WAComponent.__init__(self)
    counters := (Counter(), Counter(), Counter(), Counter(), Counter())
```

# http://127.0.0.1:8008/examples/multi

**6**

[++ =](#)

---

**3**

[++ =](#)

---

**-2**

[++ =](#)

---

**1**

[++ =](#)

---

**0**

[++ =](#)

[New Session](#) [Configure](#) [Toggle](#)  
[Halos Profile](#) [Terminate](#) [XHTML](#) 1/0  
ms

Are Components desirable

How to get components in Django?