

django



**The web framework for
perfectionists with deadlines**

Thilo Fromm, 2009



Django Web stack

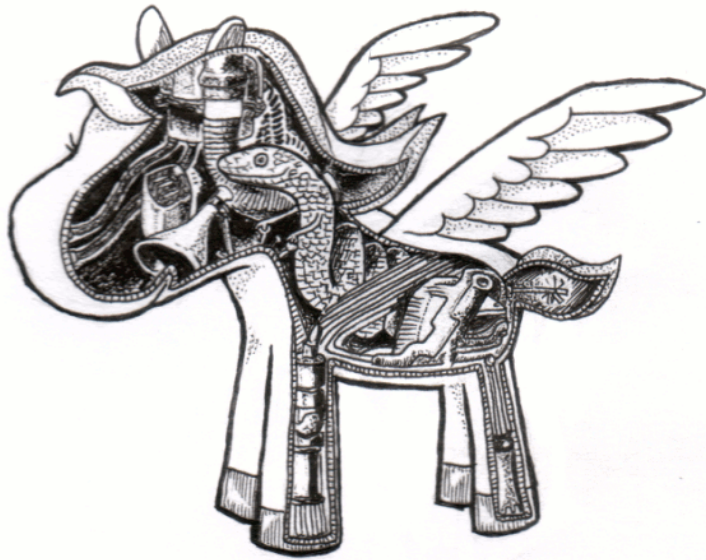


FIG. 2.
PONY "MAGIC"

State-of-the-art scripting language
Object oriented, structural, functional
Prevalently deployed
Lots of language bindings and libs
<http://www.python.org>

Python



Strongly typed – at runtime

“Duck Typing”

Forgive, don't enforce, typing

Python

Blender (parts), Poser, Maya (scripting)

Trac, ViewVC, yum

EVE online (parts, both client+server)

Google Mail, Groups, Maps backends

Youtube backend (parts)

<http://www.python.org/about/quotes/>

Software written in Python

“Django is a **high-level Python Web framework** that encourages rapid development and clean, **pragmatic design.**”

(citing <http://www.djangoproject.com/>)

Meet Django

Object-Relational Mapper

URL Dispatcher

Views, Templating System

HTML Forms abstractions

I18n, File Uploads, 1000 other little helpers

Awesome documentation

Django core framework

Admin Interface for ORM classes

Syndication (RSS)

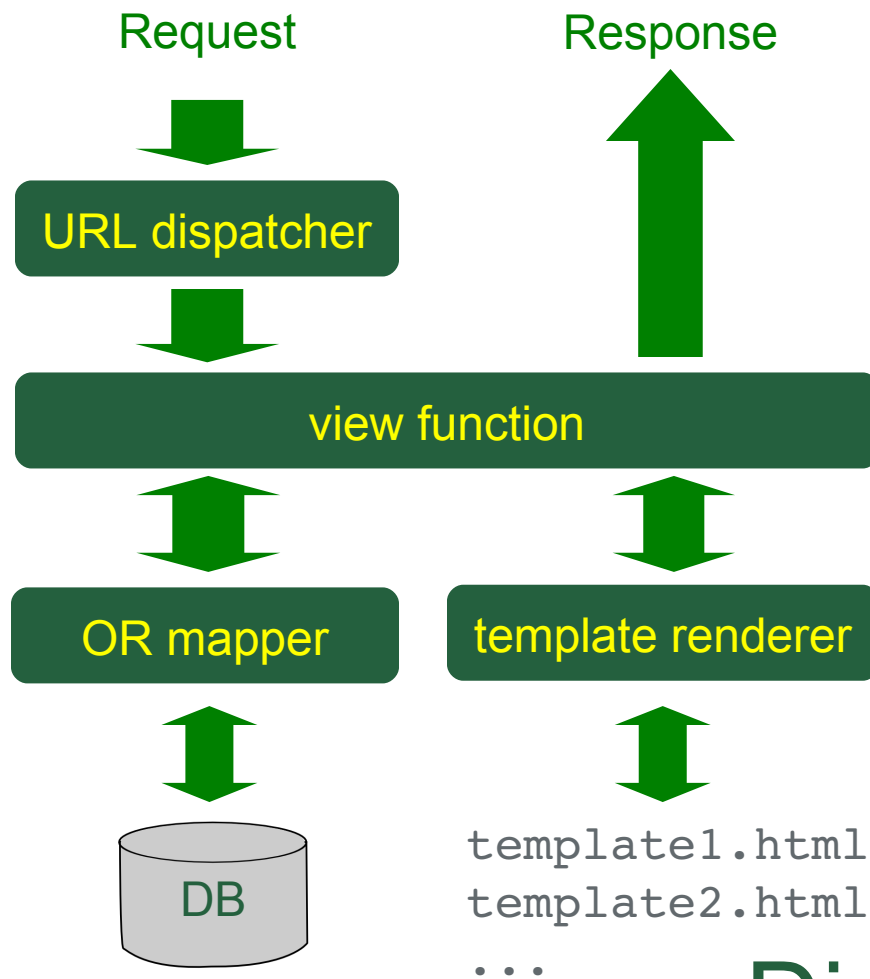
Authentication, User Management, Sessions

Caching System

Content-Types – Manager

Sitemaps Generator

Django core tools / middleware



Django Concept

A request is **URL-mapped** to a Django application, then processed in a **view**, which would fetch data using the **ORM**, and then return a rendered **template** response.”

Django Concept

```
/Everybody stand back/  
I know regular expressions
```

```
urlpatterns = patterns('',  
    (r'^reporters/(P<user>[a-zA-Z0-9]+)/publish/$',  
     views.reporter_publishes_article)  
)
```

URL mapper example

```
def reporter_publishes_article(request, user):  
    # DB query in "Reporter" table using ORM  
    r = Reporter.objects.get(name = user)  
  
    content = request.POST['article_content']  
    a = r.publish(content)  
  
    # generate response page using templates  
    t = loader.get_template('art_published.html')  
    c = Context( { 'article' : a } )  
  
    return HttpResponse( t.render(c) )
```

View method example

Object-Relational Mapping:

Define your DB tables via classes

Access data using class methods

(No SQL code required - but you can write that, too)

Extend DB classes with active code

Django OR-mapper

```
class Article(models.Model):  
    # Class attributes  
    content = models.TextField()  
    pub_date = models.DateTimeField()  
    reporter = models.ForeignKey(Reporter)
```

```
class Reporter(models.Model):  
    # Class attributes  
    name = models.CharField(max_length=70)  
    # instance methods  
    def publish(self, text):  
        a = Article( content = text,  
                    pub_date = datetime.now(),  
                    reporter = self )  
  
        a.save()  
        return a
```

OR mapper example

```
<html><body>

<h1> Success </h1>
<div> Thanks {{ article.reporter.name }},
        your article has been published at
        {{ article.pub_date }} </div>

</body></html>
```

Template example

Of course, this is only the **beginning**.

There's more!

Integrated development webserver

Integrated unit test framework

DB snapshot support (JSON- / XML-fixtures)

Design and Business Logic strictly separated

Admin Interface for manual data maintenance

Development tools

The image shows two overlapping screenshots of the Django Admin interface. The background screenshot is the 'Add book' form, and the foreground screenshot is the 'Select user to change' modal.

Add book form (background):

- URL: (Example: '/about/contact/'. Make sure to have leading slash.)
- Title:
- Content: Full HTML is allowed.
- Sites:

blogs.ljworld.com
internal.ljworld.com
www2.kusports.com
www2.ljworld.com
www.6newslawrence.com
www.6productions.com
www.lawrence.com

 Hold down "Control", or "Command" on a Mac, to select more than one.
- Advanced options: [Show Advanced options](#)
- Buttons: [Save and add another](#) [Save and continue editing](#) [Save](#)

Select user to change modal (foreground):

Search: [Go](#) [Add user](#) +

Username	E-mail address	First name	Last name	Staff status
071004				<input type="radio"/>
078asm				<input type="radio"/>
1111				<input type="radio"/>
12345				<input type="radio"/>
123bjamm				<input type="radio"/>
123kvb				<input type="radio"/>
12thstreet				<input type="radio"/>
13hawk				<input type="radio"/>
1891				<input type="radio"/>
1Q77kufan				<input type="radio"/>

Filter:

- By staff status:
 - All
 - Yes
 - No
- By superuser status:
 - All
 - Yes
 - No

Django Admin

Geo-Django: Geo data server

Django on Jython: Java Integration

pinax: Web portal applikation framework

Contact lists, User-to-user messaging, Passwort mgmt

Wiki / Blog / Commenting / Tagging / Twitter

Issue tracking, workflows

Notification framework

3rd party Add-ons

Django Debug Toolbar

Debugging / Optimization

DB migration tools for updating models

(more) Test Frameworks, Selenium-Integration

Dojo Integration with DoDjango

3rd party Add-ons

Questions?

KTHX!