

**django**

~~django~~

Without Django

applying django principles to non django projects

# I Love Django

- ▶ Using Django since the very first release
- ▶ I love the templates (so much that I rewrote the templating engine — **twice**)
- ▶ Often forcing myself **not** to use it because it does not fit.

**django**



# Solace

- ▶ Werkzeug — **WSGI**
- ▶ Jinja2 — **Templating**
- ▶ SQLAlchemy — **Database**

<http://opensource.plurk.com/solace/>

# Djangoish

views.py

```
from solace.templating import render_template
from solace.models import Tag

def tags(request):
    """Shows the tag-cloud."""
    tags = Tag.query.filter(
        (Tag.tagged > 0) &
        (Tag.locale == request.view_lang)
    ).order_by(Tag.tagged.desc()).limit(40).all()
    tags.sort(key=lambda x: x.name.lower())
    return render_template('kb/tags.html', tags=tags)
```

# Djangoish

views.py

```
from solace.templating import render_template
from solace.models import Tag

def tags(request):
    """Shows the tag-cloud."""
    tags = list(Tag.objects.filter(
        tagged__gt=0,
        locale=str(request.view_lang)
    ).order_by('-tagged')[:40])
    tags.sort(key=lambda x: x.name.lower())
    return render_template('kb/tags.html', tags=tags)
```

# Djangoish

models.py

```
class UserQuery(Query):  
  
    def active_in(self, locale):  
        ua = user_activities.c  
        return self.filter(User.id.in_(select([ua.user_id],  
                                             ua.locale == locale)))  
  
class User(RemoteObject):  
    query = session.query_property(UserQuery)  
    ...
```

# Djangoish

models.py

```
class UserManager(models.DatabaseManager):  
  
    def active_in(self, locale):  
        # not sure if that works ...  
        q = UserActivities.objects.filter_by(locale=str(locale))  
        return self.filter(user__in=q)  
  
class User(models.Model, RemoteObject):  
    objects = UserManager()  
    ...
```



# Djangoish

```
{% extends 'layout.html' %}
{% from 'kb/_boxes.html' import render_topics, render_topic_tabs %}
{% set page_title = _('Overview') %}
{% block html_head %}
{{ super() }}
<link rel="alternate" href="{% url_for('kb.overview_feed', order_by=order_by)
  %}" type="application/atom+xml">
{%- endblock %}
{% block body %}
<h1>{{ page_title }}</h1>
<div class="explanation">
  <p>{% trans -%}
    This is an overview of some of the newest topics here
    on Solace. You can view other topics grouped by
    activity, votes and hotness.
  {%- endtrans %}
</div>
{{ render_topic_tabs('kb.overview', order_by) }}
{{ render_topics(topics) }}
{{ pagination }}
{% endblock %}
```

# Djangoish

```
{% extends 'layout.html' %}
{% from 'kb/_boxes.html' import render_topics, render_topic_tabs %}
{% set page_title = _('Overview') %}
{% block html_head %}
  {{ super() }}
  <link rel="alternate" href="{{ url_for('kb.overview_feed', order_by=order_by)
    }}" type="application/atom+xml">
{%- endblock %}
{% block body %}
  <h1>{{ page_title }}</h1>
  <div class="explanation">
    <p>{% trans -%}
      This is an overview of some of the newest topics here
      on Solace. You can view other topics grouped by
      activity, votes and hotness.
    {%- endtrans %}
  </div>
  {{ render_topic_tabs('kb.overview', order_by) }}
  {{ render_topics(topics) }}
  {{ pagination }}
{% endblock %}
```

TEH SYNTAX!

# Djangoish

```
{% extends 'layout.html' %}
{% from 'kb/_boxes.html' import render_topics, render_topic_tabs %}
{% set page_title = _('Overview') %}
{% block html_head %}
{{ super() }}
<link rel="alternate" href="{{ url_for('kb.overview_feed', order_by=order_by)
  }}" type="application/atom+xml">
{%- endblock %}
{% block body %}
<h1>{{ page_title }}</h1>
<div class="explanation">
  <p>{% trans -%}
    This is an overview of some of the newest topics here
    on Solace. You can view other topics grouped by
    activity, votes and hotness.
  {%- endtrans %}
</div>
{{ render_topic_tabs('kb.overview', order_by) }}
{{ render_topics(topics) }}
{{ pagination }}
{% endblock %}
```

template tags  
rendering templates

# Djangoish

```
{% extends 'layout.html' %}
{% from 'kb/_boxes.html' import render_topics, render_topic_tabs %}
{% set page_title = _('Overview') %}
{% block html_head %}
{{ super() }}
<link rel="alternate" href="{{ url_for('kb.overview_feed', order_by=order_by)
    }}" type="application/atom+xml">
{%- endblock %}
{% block body %}
<h1>{{ page_title }}</h1>
<div class="explanation">
  <p>URL Reversing
  This is an overview of some of the newest topics here
  on Solace. You can view other topics grouped by
  activity, votes and hotness.
  </p>
</div>
{{ render_topic_tabs('kb.overview', order_by) }}
{{ render_topics(topics) }}
{{ pagination }}
{% endblock %}
```

# Djangoish

## Signals:

```
from solace.signals import after_request_shutdown, \  
    before_response_sent, after_cursor_executed  
  
# get rid of the session at the end of the request  
after_request_shutdown.connect(session.remove)  
  
# adds query debug headers for unittest if enabled  
before_response_sent.connect(add_query_debug_headers)  
  
# tracks the number of queries on the request object  
after_cursor_executed.connect(request_track_query)
```

# Why not Django?

- ▶ Nice looking code when starting the project
- ▶ Often ugly hacks to keep it running that do not look nice
  - ▶ I'm obsessed with nice looking code
- ▶ Replacing contrib apps with self written code -> not much of Django left



# Better Reasons

- ▶ not depending on `DJANGO_SETTINGS_MODULE` (might be necessary)
- ▶ if you need to be able to tear down the framework without restarting Python [`reload`]
- ▶ Data structure too complex for the Django ORM (might happen)
- ▶ You're building on non-Django code

# ... and now to something completely different

- ▶ Lamson — dispatches based on regular expressions to mail handlers
- ▶ All these templates engines looking like Django (Tornado, Liquid, all these PHP clones)

# Thanks

- ▶ Slides will be on <http://lucumr.poco.org/>