Ich

- Zope und Plone-Entwickler
- 9 Jahre
- ”Power-Entwickler”

Plones Stärken

- Warum verwende ich Plone?

Plones Stärken

- PHP
- Java
- .NET

Plones Stärken

- Enterprise
  - Beans
  - SOAP
  - ...

Plones Stärken

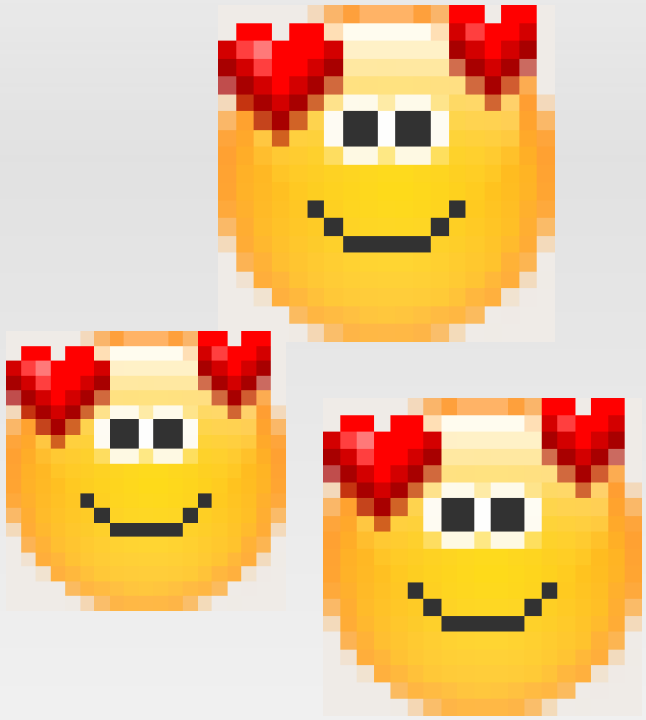
- Enterprise
  - Anzüge
  - Krawatten
  - ~~Pants~~

Plones Stärken

- Python!!1
  - fun
  - produktiv

Plones Stärken

- Python!!1
  - einfach
  - lesbar
  - expressiv

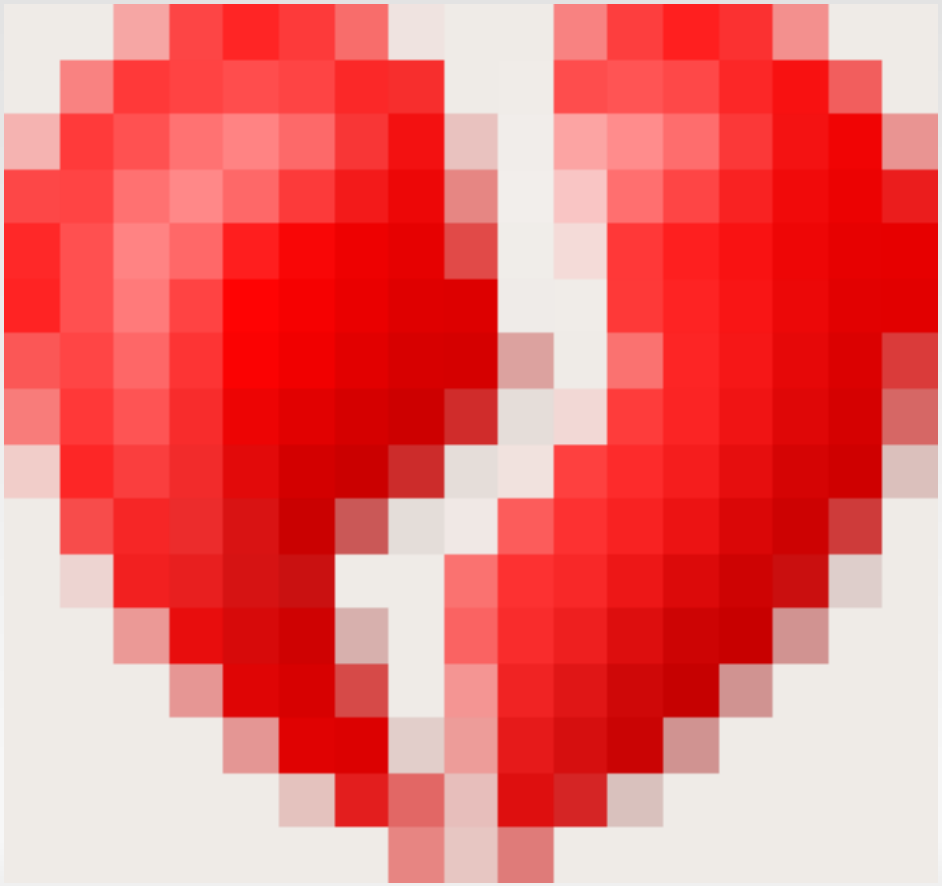


Plones Stärken

- Plone bricht Versprechen
  - WTF?
  - Plone??!

Plones Stärken

- Plone bricht Versprechen
  - ~~einfach~~
  - ~~lesbar~~
  - ~~expressiv~~
  - Love hurts



Plones Stärken

- Community!!
  - ~~Anzüge~~
  - ~~Krawatten~~
  - PANTS

Plones Stärken

- Community!!
  - international
  - freundschaftlich
  - professionell

Plones Stärken

- Flexibilität
- Erweiterbarkeit
  - Plone sagt *ja* wo andere *geht nicht* sagen

Plones Stärken

- Anpassen *through the web*
  - sehr interessant für Bastler und Communities
  - uninteressant für mich

Plones Stärken

- Plone Core hat viele Features *out of the box*

Plones Stärken

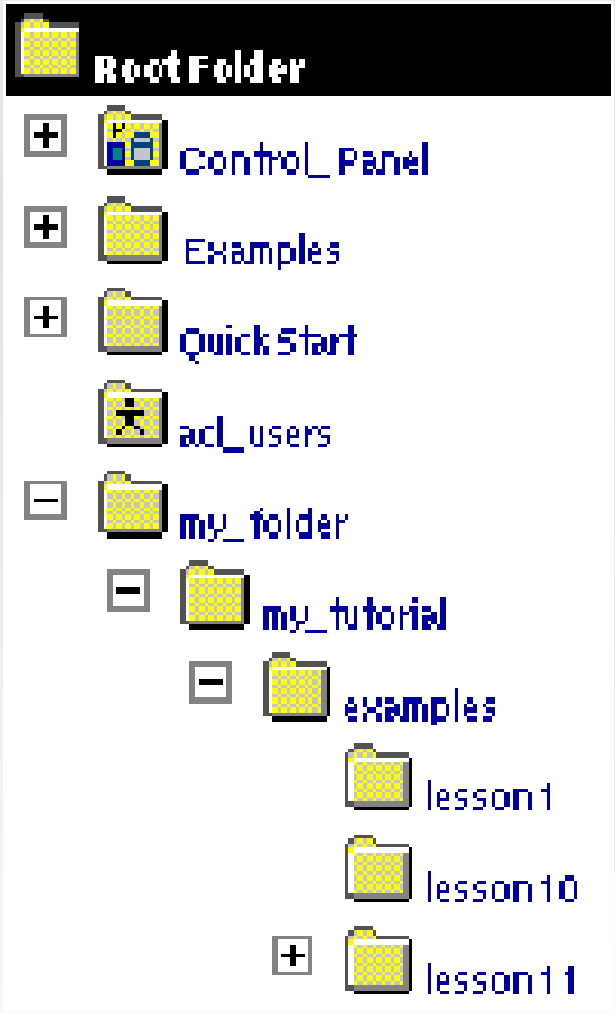
- viele freie *Add-Ons*
  - nur ein kleiner Teil gut programmiert und gewartet

Plones Stärken

- Plones *Killer-Features*
  - Object File System (OFS)
  - Rechte, Rollen, Gruppen
  - Workflows

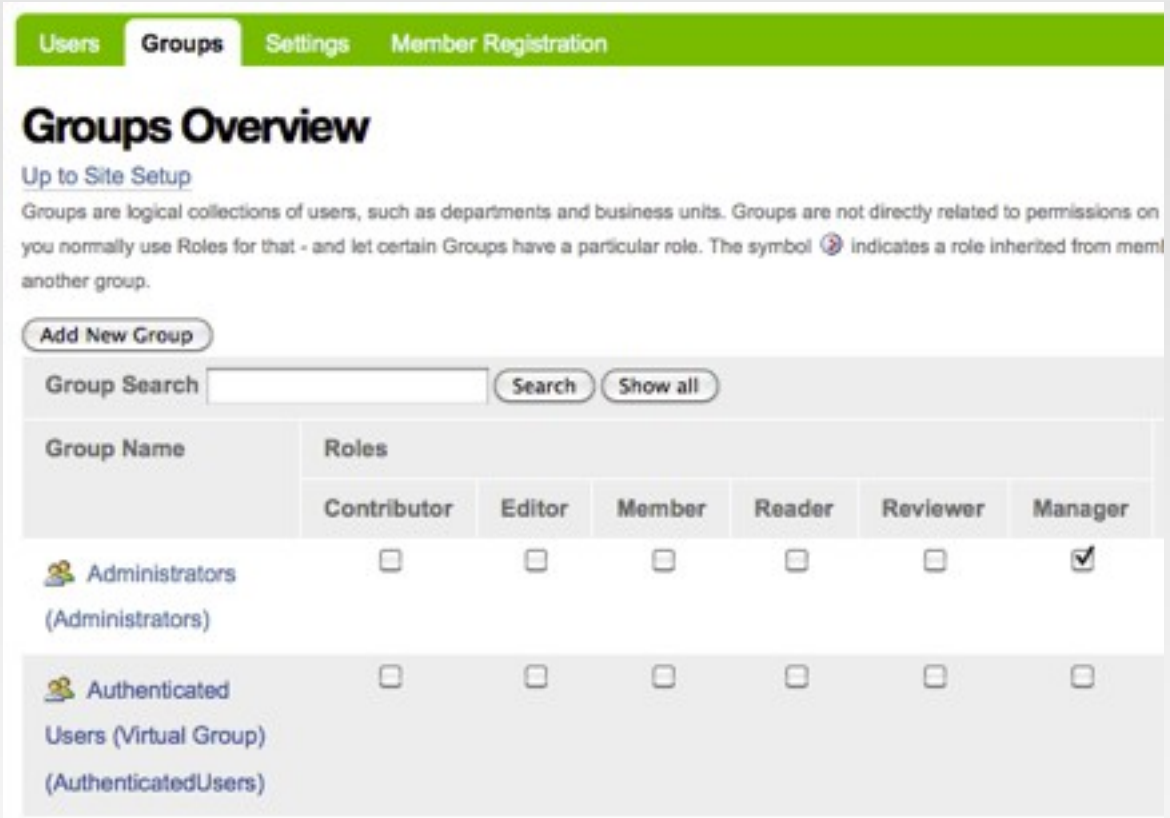
Plones Stärken

- Object File System



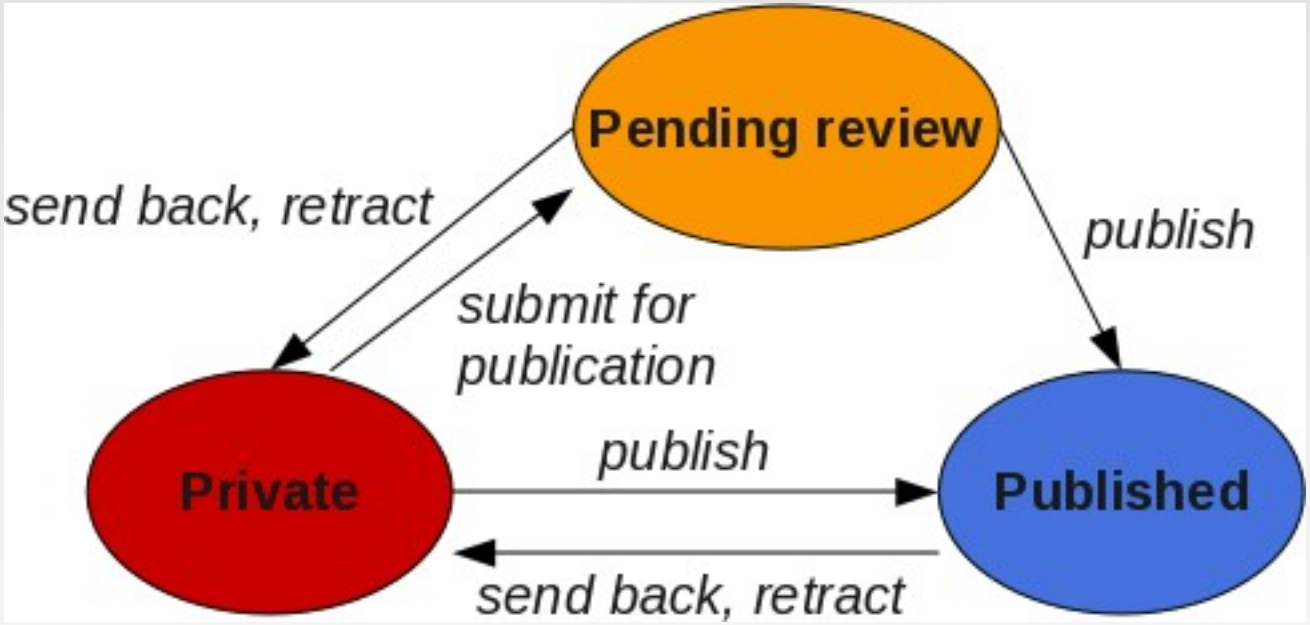
Plones Stärken

- Rechte
- Rollen
- Gruppen



Plones Stärken

- Workflows



Plones Schwächen

- Core hat **zu viele** Features *out of the box*

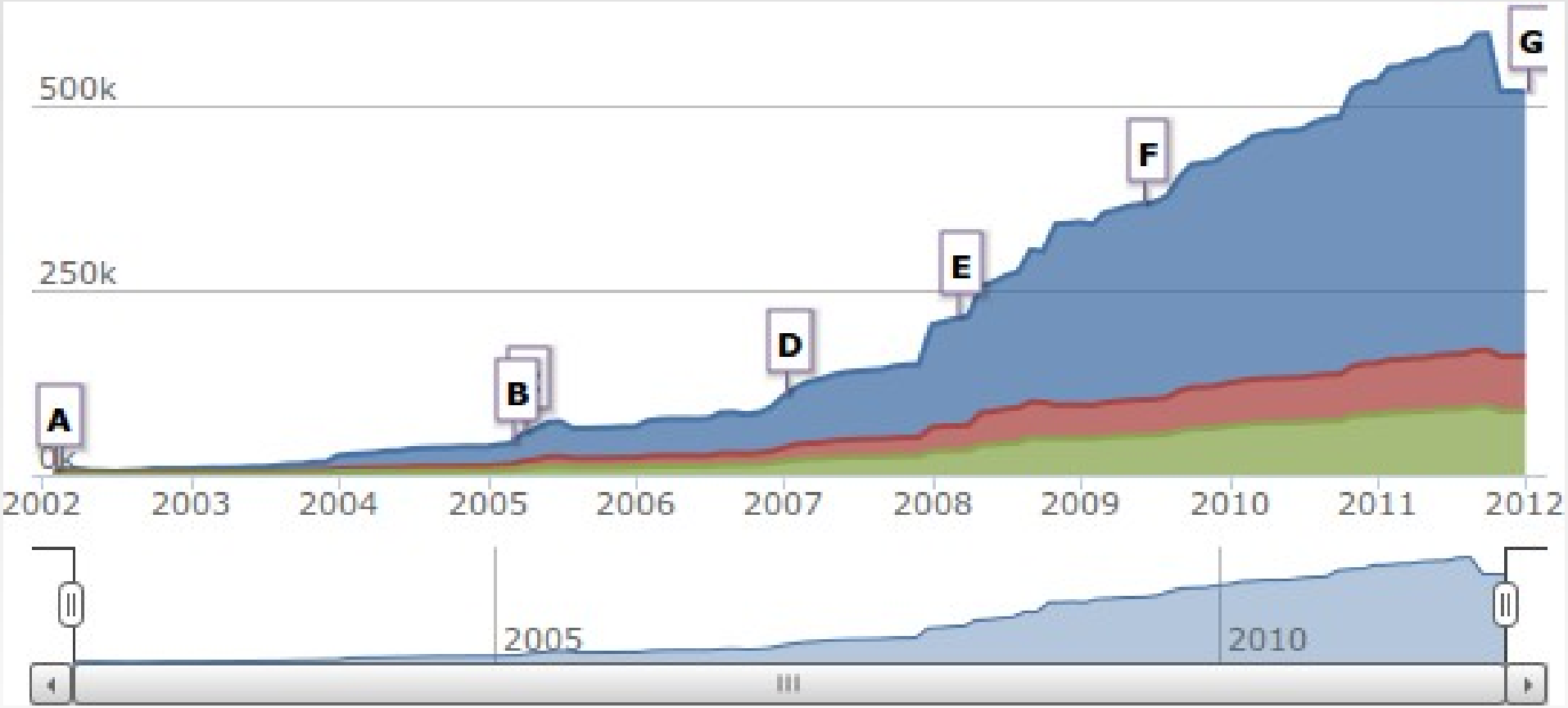
Plones Schwächen

- verschiedene konkurrierende Technologien
- *[more than] only one way to do it*

Plones Schwächen

- Code kompliziert
  - zu eifriges Adaptieren neuer Technologien
  - *Zope 3* und *Component Architecture*

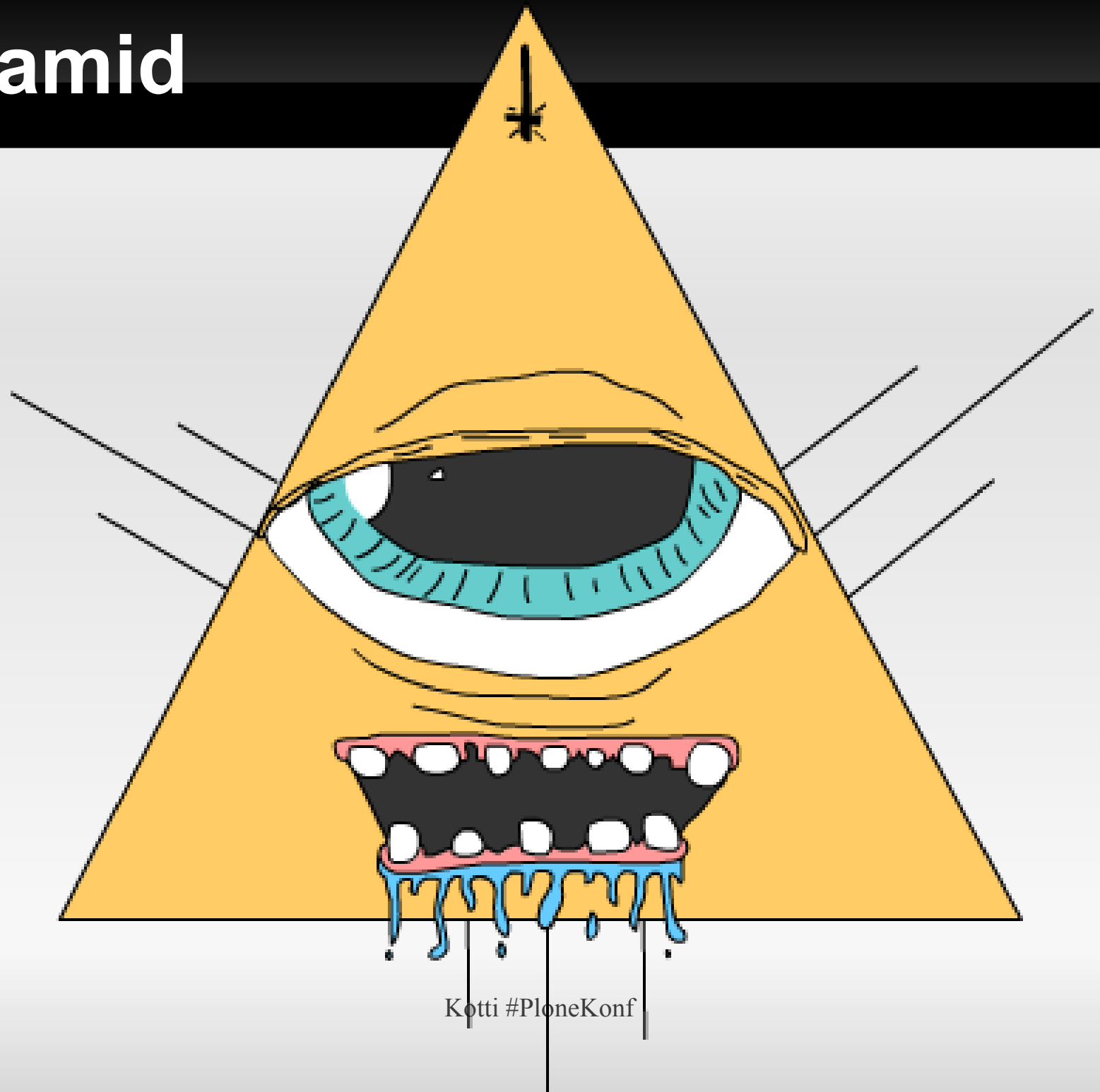
Plones Schwächen



Plones Schwächen

Plones Schwächen

Pyramid



Pyramid

- Stärken gegenüber Zope
  - *Pythonic*
  - *fits your brain*
  - Kleiner Kern
  - Ausgezeichnete Doku

Pyramid

Pyramid

Pyramid

- Basis für allerlei Anwendungen
  - nicht nur CMS
- gut für komplexe Anwendungen

Pyramid

- Zope-Wurzeln
- vgl. Plone *Killer-Features*
  - OFS (eigtl *Traversal*)
  - Rechte, Rollen, Gruppen
  - Workflows

Pyramid

- Pyramid hat viele weitere Komponenten die ein CMS braucht

Pyramid

- solider Erweiterungs-Mechanismus
- `pyramid.includes = pyramid_debugtoolbar pyramid_tm`

Pyramid

- Internationalisierung: `pyramid.i18n`

Pyramid

- *Templates* und *Customization*: `Chameleon` (ZPT) `pyramid.asset`

Pyramid

- Rechte, ACLs, Gruppen: `pyramid.security`

Pyramid

- keine Anwendung
- kein CRUD
- keine Datenbank
- *unopinionated*

Pyramid

- Ich will mein CMS mit Pyramid machen!
- *WHAT DO?*

Pyramid

- für unser CMS fehlt noch
  - Verdrahtung
  - Konfiguration

Pyramid

- für unser CMS fehlt noch
  - Persistenz
  - OFS
  - User Interface
  - Formulare

Pyramid

- für unser CMS fehlt noch
  - Benutzer / Benutzerverwaltung
  - *Control Panel*
  - Schnittstelle für *Add-Ons*

Kotti

- genau diese Dinge liefert

Kotti ♥ Pyramid

- schließt die Lücke zw. Pyramid und CMS
- ist *Framework* und *out of the box* CMS, wie Plone

Kotti ♥ Pyramid

- nutzt bestehende Pyramid-Mechanismen voll aus
- und Komponenten aus Pyramid- und Python-Community

Kotti ♥ Pyramid

- wenig neue Konzepte
- Pyramid-Entwickler sollen möglichst Bekanntes vorfinden
- erbt Pyramids Stärken

Kotti ♥ SQLAlchemy

- Kottis Datenbank
  - verwendet SQL Datenbank
  - nicht ZODB
  - SQLAlchemy

Kotti ♥ SQLAlchemy

- Kottis Datenbank
  - keine großen Abstraktionen
  - SQLAlchemy API
  - abgesehen von Nodes

Kotti ♥ Twitter Bootstrap

- *Default* Oberfläche benützt *Bootstrap*
  - CSS, JavaScript
  - *Responsive Design*
  - dokumentiert, getestet

Kotti ♥ Twitter Bootstrap

Kotti ♥ Colander und Deform

- *Default* Formulare basieren auf *Deform*
  - Schemas
  - Validierung
  - Widgets

Kotti ♥ Beaker

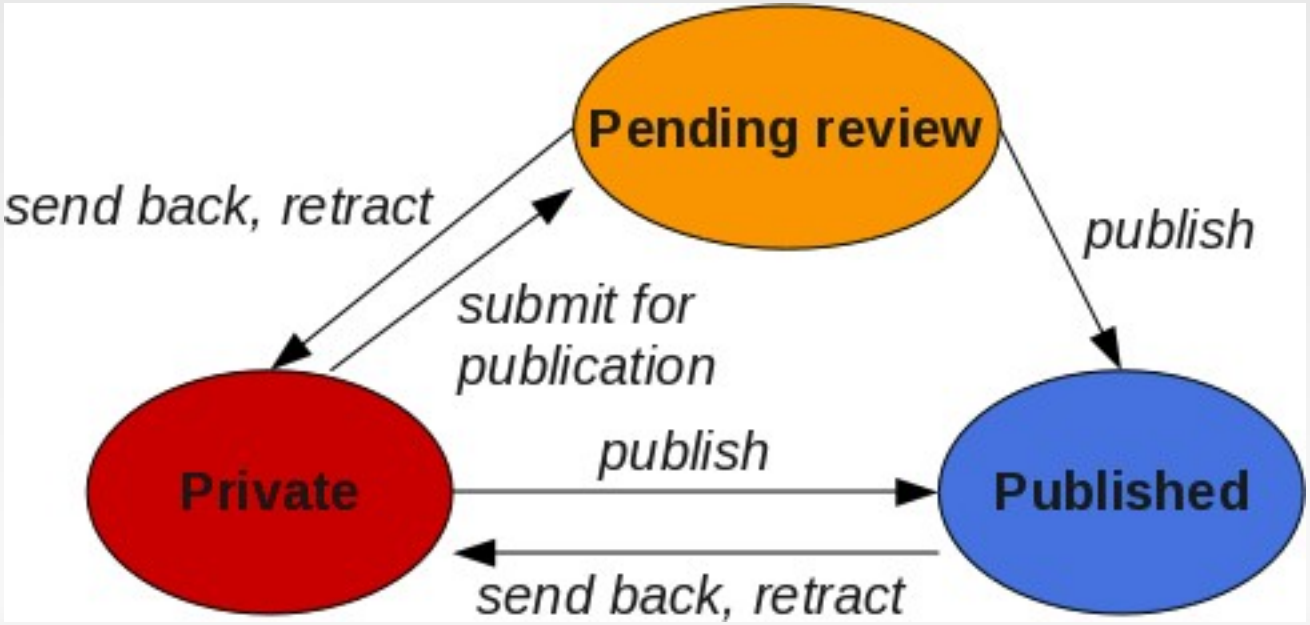
- Kotti verwendet *Beaker* für das *Sessioning*
  - Daten die vorübergehend gespeichert werden
  - konfigurierbar

Kotti ♥ plone.i18n

- Plone??!
- Ja, `plone.dott`!

Kotti ♥ repoze.workflow

- leicht zu integrieren
- nicht *out of the box* dabei



Kotti ♥ YOU

- Kotti konfiguriert alle diese Komponenten
- damit du es nicht machen mußt
- *Batteries included*

Kotti ♥ YOU


I ♥ Kotti
Steigende Mieten am Kottbusser Tor stoppen!
Kottbusser Tor'da yükselen kiraları durdurun!
Kotti & Co / Die Mietergemeinschaft am Kottbusser Tor
kottico@gmx.net / Facebook: kotti Undco

Kottis Design

- möglichst einfach
- kein *Big Design*
- konzentriert sich auf:
  - (kleinen) Core
  - Erweiterbarkeit

Kottis Nodes API

- `root = get_root()`
- *dict-like*
  - `root['hey'] = Document(...)`
  - `root['hey'].title = u'Ho'`
  - `del root['hey']`
  - `root.values() # usw.`

Kottis Nodes API

- Reihenfolge (*ordered*)
  - `child = root.children.pop(1)`
  - `root.children.insert(0,child)`
- alle Content-Typen (Dokumente, Files, …) sind Nodes

Kottis Benutzerdatenbank API

- `P = get_principals()`
- ```
  P['daniel'] = {
        'name': 'daniel',
        'title': 'Daniele',
        'password': 'woops',
        }
  ```
  ```

Kottis Schnittstellen für Add-Ons

- `kotti.includes`
- `kotti.populators`
- `kotti.available_types`
- `kotti.asset_overrides`

Kottis Templates

- Master-Templates
- Unterschied zwischen
  - Öffentliche Oberfläche
  - Redakteurs-Oberfläche

Kottis Templates

- leicht austauschbar
  - z.B. nur öffentliche Oberfläche
  - oder nur Dokument-Template

Kottis Templates

- Kotti Slots (Viewlets)
  - ```
    def render_hello(context, request):
        return u'Hello, World!'

    from kotti.views.slots import *
    register(
      RenderLeftSlot, None, render_hello)
    ```
    ```

Kottis Zukunft

Kottis Vergangenheit

- existiert seit > 1 Jahr
- ich verwende es
  - ein großes Projekt
  - ein kleines Projekt
  - andere Projekte, Websites

Kottis Vergangenheit

- *Add-Ons*
  - kotti_calendar
  - kotti_twitter
  - kotti_contactform
  - kotti_analytics

Kottis Gegenwart

- Sprint!

  - *WYSIWYG* Hochladen von Dateien und Bildern
  - `kotti_blog`
  - `kotti_whatever`

Ciao

- github.com/Pylons/Kotti
- kottidemo.danielnouri.org
- #kotti auf Freenode
- @dnouri
- daniel.nouri@gmail.com