



= pdfcomment =

- upload a pdf file
- have it splitted page by page
- comment rectangular areas.

Author: Marco Paolini - Credra S.r.l.



= Problem =

Our customer is a newspaper publisher.

His editors use specific tools like
Qua...press to create page layouts.

The authors are journalists, they can't
use these tools but ***still*** need a way
to interact with the editing process.



= Solution =

1 the editors ***upload*** the paper draft

2 the author ***comments*** it

3 the editor makes the correction

4 goto 1

== other features ==

- versioning support
- email notification when:
 - a new comment is submitted
 - a new paper version was uploaded
- it is the customer's website!



= Architecture (recipe) =

- 1 django app
- 4 format converters:
 - gs (pdf->png)
 - PIL (jpeg->png & thumbnails)
 - pygmentize (UTF8 sources->png)
 - ImageMagick (strips whitespaces)
- 2 javascript libs:
 - jquery
 - lightbox
- some javascript w/AJAX hacks
- 1 relational db: Sqlite
- lighttpd and mod_magnet



= upload a paper and split it =

implemented in save method of Paper
class

creates Page objects that contain image
representation of each page.

== pluggable ==

- .py
- .jpeg
- .png

link: add a paper

link: see it splitted



= COMMENT A PAGE =

- comment area is a 20% opaque <div>
- created on mouseclick
- can be moved around & resized
- form for async POST
- listing of all previous comments

link: comment a page



= where's my comment now? =

It's in the database, of course

it is a row in table Comment

link: comment a page



= really simple versioning =

link: upload 2nd draft

- comments belong to first draft
- closed commenting on first draft



= deployment I =

```
lighttpd--mod_magnet
|      |
|      |
|      -----fcgi-----django
|                                     |
sendfile                             |
|                                     |
|                                     --- sqlite
|                                     ---  gs
media                                --- pygments
files
(images, css, js)
```

(thanks to flup for fcgi)



= deployment II =

- lighttpd
 - embeds a scripting language (LUA) for request/response manipulation
 - useful for cache-related issues
- use popen for image operations
 - reduces memory leaks
 - fewer LOCs required
 - more resistant to memory exhaustion
- separate code&data
 - django app installed as a package
 - django database on a separate dir
 - lighttpd chrooted



= pdfcomment eats dogfood =

- let's upload some pdfcomment code
- see it pygmentized
- comment it!

link: [django admin](#)

link: [user area](#)



= Effort =

< 2K slocs including tests

py: 70%

html: 10%

css: 10%

js: 10%

test using django test suite and
BeautifulSoup

marco@credra.com

