



WOK

or

Web Objects Kitchen

a web development framework

(sort of.)

WHY?

- In 1999 we were using Python (Zope 2)
- In 2004 we were using Python (Twisted/Nevow)
- In 2005 we were pushing Python to its limits...
- ...or was Python pushing us to ours?

“If you can't code it in Perl, why don't you try C#?”

– lupus, at webb.it 2004

- # ASP.NET
- C# (2.0) seemed to be a great language
 - At that time ASP.NET was the only option, but:
 - No support for server-side DOM
 - Made it easy to generate invalid HTML
 - URL mapping to code (“routing”) was inexistent
 - Bad support for multiple themes and languages
 - Templates allowed for lots of side-effects
- # NOT

WOK 1

No plan (bad!) but we did know what we wanted:

- multiple languages (cookies or headers or URLs)
- themes: pages are built out of localized XML templates and in-code DOM fragments)
- flexible URL routing: a single resource (URL) is mapped to multiple *components* that collaborate to build the page
- page transactions



Torino prende il volo



WORLD AIR GAMES | EVENTO | SPONSOR | VOLONTARI | AREA MEDIA | CONTATTI | FUN AREA

3 Febbraio 2011

ENGLISH

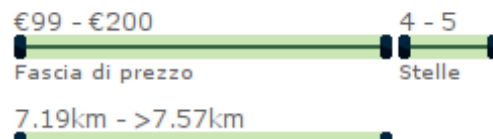
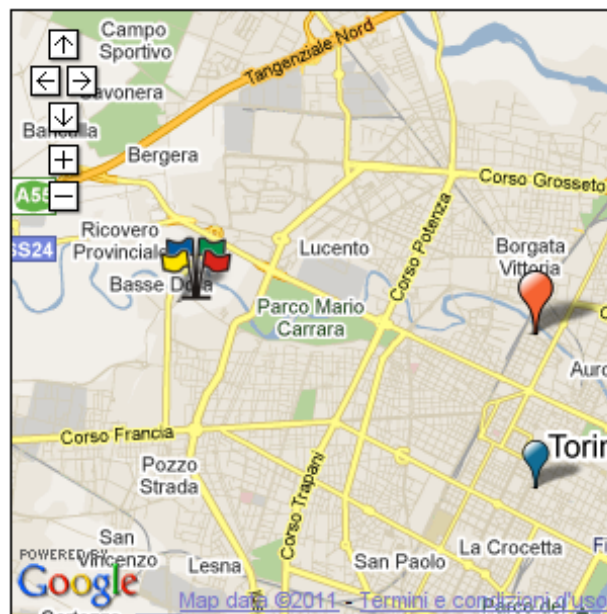
DISCIPLINE:

- » Acrobazia
- » Aeromodellismo
- » Deltaplani
- » Elicotteri
- » Experimental
- » Mongolfiere
- » Paracadutismo
- » Parapendio
- » Ultraleggeri
- » Volo a vela

Calendario gare

WORLD AIR GAMES TORINO 2009

Accreditation & Accommodation



Soggiorno, inserisci le tue date di arrivo e di partenza desiderate

Arrivo	Partenza	Adulti	<input type="button" value="Cerca"/>
<input type="text" value="03/02/2011"/>	<input type="text" value="04/02/2011"/>	<input type="text" value="1"/>	

Quotazioni e disponibilità per 1 notti.

Ordina: **Prezzo** | [Nome](#) | [Categoria](#) | [Distanza](#)

Pacific Hotel Fortino **** Città, 7,19 km da



Doppia uso singola standard Tariffa - Pernottamento e prima colazione **€99,00** [prenota](#)

Golden Palace ***** LUX Centro Città, 7,57 km da



**Then
We
Were
Happy!**



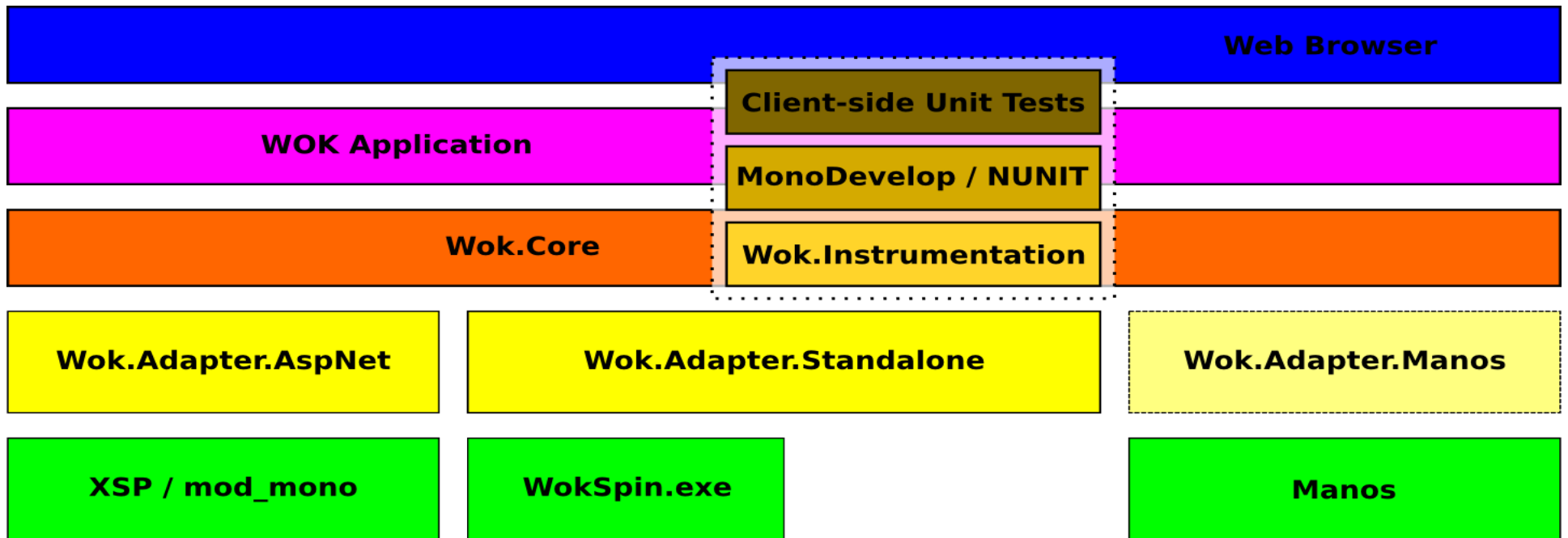
WOK-2

In 2009 we started WOK-2 development (and this time, we had a plan):

- abstract WOK application from hosting
- “componentize” everything
- MonoDevelop support
- stuff from WOK-1: templates, skinning, server-side DOM, page transactions, data validation.

Yes, I now that everytime one writes “componentize” a kitten is killed but I quoted it, OK?

LAYERS



FELLOW

What happens on incoming HTTP requests?

- “URL processing” - extract information
- “URL mapping” - register components
- “Resource running”
 - Initialize registered components
 - Run resource workflow using events
 - Send result back to client

APP

```
<app xmlns="http://initd.org/ns/wok-1.0"
  xmlns:wokc="clr-namespace:Wok.Core;assembly=Wok.Core"
  xmlns:wokp="clr-namespace:Wok.Core.Processors;assembly=Wok.Core"
  xmlns:profiling="clr-namespace:WokRecipes.Profiling;assembly=WokRecipes.Profiling"
  xmlns:app="clr-namespace:VentanaMice.Web;assembly=VentanaMice">
  <authentication-managers>
    <profiling:AuthenticationManager CookieName="ventana.mice.auth"
      CookieKey="df096339d64e7cee4ecb357726539647"/>
  </authentication-managers>
  <url-processors>
    <wokp:ActionView/>
    <wokp:Rest/>
    <wokp:VirtualHost/>
  </url-processors>
  <url-mappers>
    <app:Mapper/>
  </url-mappers>
</app>
```

CONF

- You can have as many mappers as you like.
- A builtin declarative XML-based mapper:

```
<map xmlns="http://initd.org/ns/wok-1.0" ...>  
  <path pattern=".*" mode="ContinueOnMatch">  
    <woku:SkinService DefaultTheme="tutorial"/>  
  </path>
```

...

```
<path pattern="^(.+)$">  
  <wokc:StaticFile Path="$1"/>  
</path>
```

```
</map>
```



The Final Struggle for the Evolution of Taste Become a Guru of Coolhunting



Prey Coolhunters Search

fog

OBSERVER
Rank: 90

28%

My Fans 3

My Fav Coolhunters 7

My Prey 2

Catch a Prey!

Where is the Cool?



- Latest
- Top Prey
- Top Gurus
- Trendboxes
- Trenditory
- Wall
- Backpack

Top Prey



Latest Prey

- Alee**
0 votes
What do you think
- Alee**
0 votes
Taylor Swift Your not sorry
- Alee**
0 votes
Taylor Swift Fearless I <3 TAYLOR SWIFT
- Alee**
0 votes
Sneakers with light up laces
- Alee**

PLUG!

Components manage everything, from database connections to HTTP response construction. We have many builtin components:

- HtmlTemplate, JsonTemplate, ...
- StaticFile, DynamicFile, StreamingFile
- WokDbConnection, ExceptionSkin, ...

Recipes are collections of components and a mapper, e.g., login, change/recover password, manage account.

RUN!

Components register for Resource events during their Initialize() phase. And here are the events:

- TransactionBegin
- Authorize (component can redirect)
- Prepare
- Validate
- Process (component can redirect)
- TransactionEnd/TransactionAbort
- Render (no writes here)
- SendPage

CODE

Features better demoed in code:

- File system skinning
- Declarative templates with data stacks
- GET & POST data validation
- Data proxies



`git://luna.dndg.it/public/wok-2`

`<fog@initd.org>`

Currently working on:

- MonoDevelop integration
- Component caching
- Profiling, Reporting and CMS recipes