

No Act on ACT

Stefan Hornburg (Racke)

2016-09-21

Contents

Conferences	4
YAPC::EU 2008 in Copenhagen	4
PerlDancer for Perlers (FOSDEM 2011)	4
Perl Dancer Conference 2014	4
Using ACT in 2013/2014	6
Modern Perl for web applications	7
mod_perl vs PSGI	7
Apache version	7
Dancer2 Advantages	7
Manual SQL vs DBIx::Class	8
More Reasons for "No act on ACT"	9
Single Point of Failure	9
Development	9
Hackathon in Lyon	9
Responsiveness	9
Perl Dancer Conference Website	12
Web Application Features	13
Maps with Leaflet	13
Venue on Frontpage	13
Leaflet Library	14
Leaflet Code	14
User Map	14
Cropping photos	14
Current photo	15
Ballerina	15
Cropping I	15
Cropping II	15
Final Result	15
Jerop Library	15
SEO / Modern Web Design	15
Secure site with HTTPS	16
Speaking Links I	16
Speaking Links II	16
Administration	16
Navigation	20
Development	21
Dancer2::Debugger	21

DBIx::Class Techniques	22
Truth of ResultSet	22
Chaining	22
Relationship Traversal	22
Hashref Inflator	23
Predefined Searches	23
Dancer2	24
Code Structure	24
Main Module	24
Route module I	24
Route module II	25
Hooks used	25
Plugins used	25
Plack Middleware	25
Cross-site request forgery (CSRF/XSRF)	27
Summary	28
Modern Perl	28
Missing features	28
Perl Dancer Conference 2016	28
Call for developers	28
Questions	28

Conferences

YAPC::EU 2008 in Copenhagen



PerlDancer for Perlors (FOSDEM 2011)

Sawyer's presentation

Perl Dancer Conference 2014

<http://act.perl.dance/eic2014/>

PHP – the nonsensical



Using ACT in 2013/2014

- Managed software
- Listed in other conferences (Marketing)
- User database

Modern Perl for web applications

- PSGI / Plack
- Web application framework (Dancer2)
- Web framework (Bootstrap, JQuery)
- DBIx::Class
- Moo and friends
- Testing

mod_perl vs PSGI

- Apache / mod_perl
 - slow
 - memory footprint
 - deployment related code
- PSGI
 - runs on any webserver supporting PSGI
 - fast (Nginx)
 - simple configuration (Nginx)
 - no deployment related code

Apache version

Apache/1.3.42 (Unix) mod_perl/1.31

more than 6 years old

Dancer2 Advantages

- Clean OO interface
- Routes
- Hooks
- Engines
- Multiple Applications

- Web application
- REST API
- Admin interface

Manual SQL vs DBIx::Class

- Manual SQL
 - error prone
 - compatibility problems
 - less secure
- DBIx::Class
 - Business logic
 - works on dozens of RDBMS
 - usually faster
 - more secure

More Reasons for "No act on ACT"

- Single Point of Failure
- Development
- Responsiveness
- No `https://`

Single Point of Failure

- Baltimore Perl Workshop
- Eventbrite

Development

- Last commit on ACT: July 2015 `https://github.com/book/Act`
- Last commit on ACT in the box: September 2015

Hackathon in Lyon

Responsiveness





Picture by Guillaume Brocker, CC BY-SA 3.0

Perl Dancer Conference Website

PERLDANCER 2016

VIENNA, AUSTRIA

SEPT 20 - SEPT 22

<https://github.com/interchange/Perl-Dancer-Conference>

Web Application Features

- Maps with Leaflet
- Cropping Photos
- Administration
- SEO / Modern Design
- Dancer2::Debugger

Maps with Leaflet

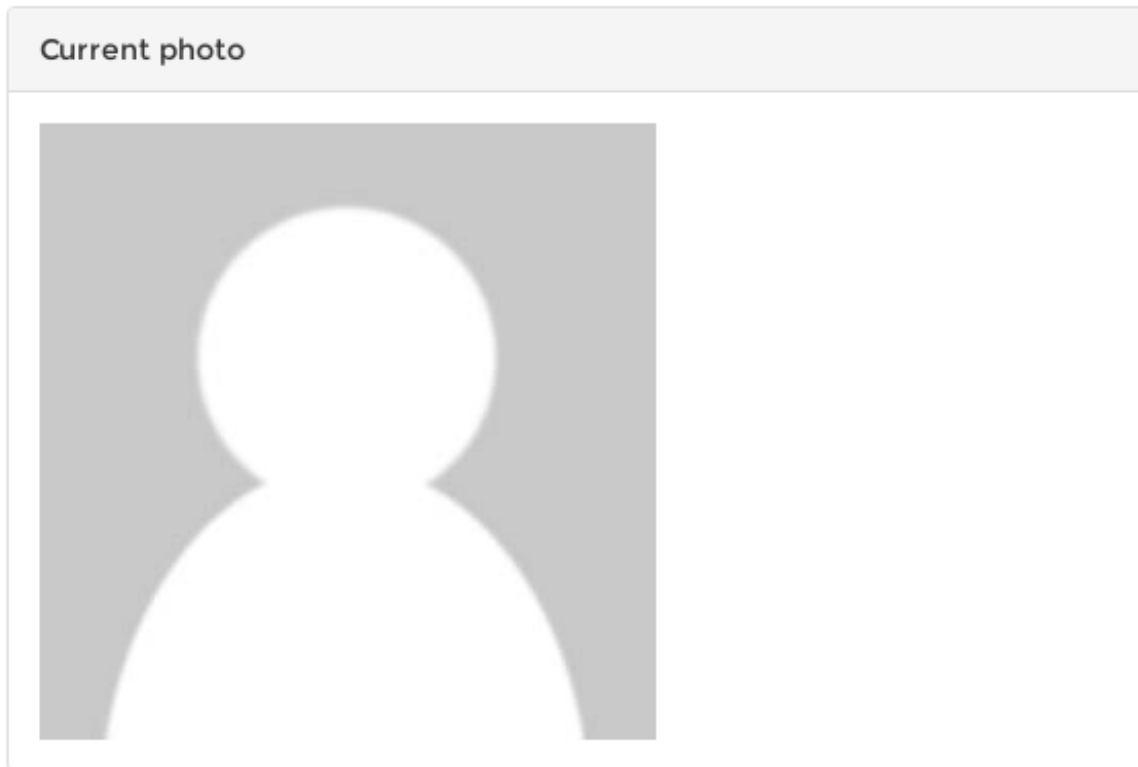
- Venue on Frontpage
- Leaflet library
- Leaflet code
- User map

Venue on Frontpage

Conference Venue - Hotel Schani Wien 21/22 Sep
📍 Karl-Popper-Straße 22, 1100 Vienna, Austria



Current photo



</profile/photo>

Ballerina

Cropping I

Cropping II

Final Result

Jcrop Library

Jcrop - jQuery Image Cropping Plugin

SEO / Modern Web Design

- Secure site with HTTPS
- Speaking links



Secure site with HTTPS

- More secure and gives users more confidence
- Small ranking boost with Google
- Free certificates with Let's Encrypt

Speaking Links I

Speaking Links II

ACT <http://act.yapc.eu/ye2016/talk/6780>

PDC <https://www.perl.dance/talks/40-no-act-on-act>


Administration

- Users
- Talks
- Tickets
- Events
- News

Upload a new photo min 300x300px, max 10MB



 ballet-1033163_1280.j ...

 Remove

 Upload

 Browse ...

Please crop image



Submit





No act on ACT

Navigation

PERLDANCER 2016 TALKS ▾ TICKETS WIKI USERS ▾ SPONSORS ▾ ADMIN ▾ MY ACCOUNT ▾

Name	Type	Scope	URI	Description	Alias	Priority	Active	Edit	Delete	Child
▼ Admin	nav	menu-admin	admin	Administration		0	Yes	✎	✖	👤
Navigation	nav		admin/navigation	Navigation Administration		0	Yes	✎	✖	👤
Talks	nav		admin/talks	Talks Administration		0	Yes	✎	✖	👤
Users	nav		admin/users	User Administration		0	Yes	✎	✖	👤
T-shirts required	nav	menu-admin	admin/t-shirts			0	Yes	✎	✖	👤
Events	nav	menu-admin	admin/events	Event Admin		0	Yes	✎	✖	👤
Tickets Sold	nav	menu-admin	admin/tickets			0	Yes	✎	✖	👤
News	nav		admin/news	News Administration		0	Yes	✎	✖	👤
▼ Talks	nav	menu-main				60	Yes	✎	✖	👤
Talks	nav		talks	List of talks		100	Yes	✎	✖	👤
Submit a talk proposal	nav		talks/submit			90	Yes	✎	✖	👤
Favourite Talks	nav		talks/favourite			80	Yes	✎	✖	👤
Talks Schedule	nav		talks/schedule			70	Yes	✎	✖	👤
Personal Schedule	nav	menu-main	myschedule			68	Yes	✎	✖	👤
Speakers	nav		speakers	Meet Our Speakers!		65	Yes	✎	✖	👤
iCal export	nav		talks/timetable.ics			60	No	✎	✖	👤
2015 Talks Archive	nav	menu-main	talks/archive/2015	Perl::Dancer Conference 2015 Talks Archive		30	Yes	✎	✖	👤
Tickets	nav	menu-main	tickets	Get Your Seat Before It's Too Late		40	Yes	✎	✖	👤
Wiki	nav	menu-main	wiki	Wiki		30	Yes	✎	✖	👤
▼ Users	nav	menu-main				25	Yes	✎	✖	👤
Map	nav		users	User Map		100	Yes	✎	✖	👤
Search	nav		users/search	User Search		80	Yes	✎	✖	👤
Statistics	nav		users/statistics	User Statistics		60	Yes	✎	✖	👤
▼ Sponsors	nav	menu-main				20	Yes	✎	✖	👤
Our Sponsors	nav		sponsors	Sponsors		100	Yes	✎	✖	👤
Diamond Sponsor	nav		sponsors/diamond			50	Yes	✎	✖	👤
Gold Sponsor	nav		sponsors/gold			40	Yes	✎	✖	👤
Silver Sponsor	nav		sponsors/silver			30	Yes	✎	✖	👤
Bronze Sponsor	nav		sponsors/bronze			20	Yes	✎	✖	👤
Sponsoring	nav		sponsoring	Become a Sponsor!		80	Yes	✎	✖	👤
▼ Login	nav	top-login				10	Yes	✎	✖	👤
Login	nav		login	Login or Register		100	Yes	✎	✖	👤
Register for Conference	nav		register			80	No	✎	✖	👤
Reset Password	nav		reset_password			70	Yes	✎	✖	👤

Development

Dancer2::Debugger

Dancer2::Debugger Presentation



📅 Sep 22, 2016
🕒 11:30 - 12:10
📍 Schani
👤 [Peter Mottram](#)

Plack::Debugger meets Dancer2

I've been a user of Plack::Middleware::Debug with its various Dancer panels since I started working with Dancer in early 2014. After seeing Stevan Little's presentation of his newly released Plack::Debugger at LPW2014 I decided I needed to play with this great tool ASAP...

Sixteen months later in April 2016 I finally started using Dancer2 for real work and found that no debug panels existed for Plack::Middleware::Debug so I looked for Plack::Debugger panels for Dancer2 and found nothing. The result was that I started work on Dancer2::Debugger which provides Dancer2 panels for Plack::Debugger along with a Dancer2 plugin to make configuration as simple as possible.

This talk will cover aspects of the design and use of Plack::Debugger and its companion Dancer2::Debugger along with practical demos.

DBIx::Class Techniques

- Set Based DBIx::Class <http://www.perladvent.org/2012/2012-12-21.html>
 - Truth of ResultSet
 - Chaining
 - Relationship traversal
 - Predefined searches
- Hashref Inflation

Truth of ResultSet

This doesn't execute SQL:

```
my $talks = $schema->resultset('Talk')->search(...);
```

This does:

```
my $first_talk = $schema->resultset('Talk')
    ->search(...)->first;
```

Chaining

```
my $talks = $schema->resultset('Talk')
    ->search({-bool          => 'accepted'})
    ->search({conferences_id => 1})
    ->search({room           => { '!=' => '' }},
    ->search({start_time     => {
        '>=' => '2015-10-22 00:00:00'
        '<=' => '2015-10-23 00:00:00'
    }}),
);
```

Relationship Traversal

```
# countries dropdown
$tokens->{countries} = [
    $users->search_related(
        'addresses',
        {
            type => 'primary',
        },
    ],
```

```

)->search_related(
  'country', undef,
  {
    columns => [ 'country_iso_code', 'name' ],
    distinct => 1,
    order_by => 'country.name',
  }
)->hri->all
];

```

Hashref Inflater

```

# countries dropdown
$tokens->{countries} = [
  rset('Country')->search( undef,
    { columns => [ 'country_iso_code', 'name' ],
      order_by => 'name' } )
  ->hri->all
];
unshift @{$tokens->{countries} },
  { country_iso_code => undef, name => "Select Country" };

```

Predefined Searches

User class:

```

sub last_conference_attended {
  my $self = shift;
  my $conference = $self->conferences->search({}, {
    order_by => { -desc => 'start_date' },
  })->first;
}

```

Web application:

```

my $conference = $user->last_conference_attended;

```

Dancer2

- Code structure
- Hooks
- Plugins
- Plack::Middleware

Code Structure

- Main module
- Multiple modules for routes

Main Module

```
use Dancer2;

use PerlDance::Routes::Account;
use PerlDance::Routes::Admin;
use PerlDance::Routes::Data;
use PerlDance::Routes::PayPal;
use PerlDance::Routes::Profile;
use PerlDance::Routes::Survey;
use PerlDance::Routes::Talk;
use PerlDance::Routes::User;
use PerlDance::Routes::Wiki;
```

Route module I

```
use Dancer2 appname => 'PerlDance';

use Dancer2::Plugin::Auth::Extensible;
use Dancer2::Plugin::DataTransposeValidator;
use Dancer2::Plugin::DBIC;
use Dancer2::Plugin::Email;
use Dancer2::Plugin::Deferred;
use Dancer2::Plugin::Interchange6;
use Dancer2::Plugin::TemplateFlute;

prefix '/profile';
```


Route module II

```
get '/talk/create' => sub {
  my $tokens = {
    title => "New Talk",
  };

  $tokens->{form} = form('create-update-talk');
  $tokens->{form}->reset;
  $tokens->{form}->fill( duration => 40 );

  add_durations_token($tokens);

  template 'profile/create_update_talk', $tokens;
};
```

Hooks used

```
hook 'before_template_render' => sub {
  my $tokens = shift;

  if ( my $user = logged_in_user ) {
    $tokens->{logged_in_user} = schema->current_user;
  }

  $tokens->{conference_name} = setting('conference_name');
};
```

Plugins used

Dancer2::Plugin::Auth::Extensible User Authentication

Dancer2::Plugin::DataTransposeValidator Form validation

Dancer2::Plugin::DBIC DBIx::Class Wrapper

Dancer2::Plugin::Deferred Flash messages after redirect

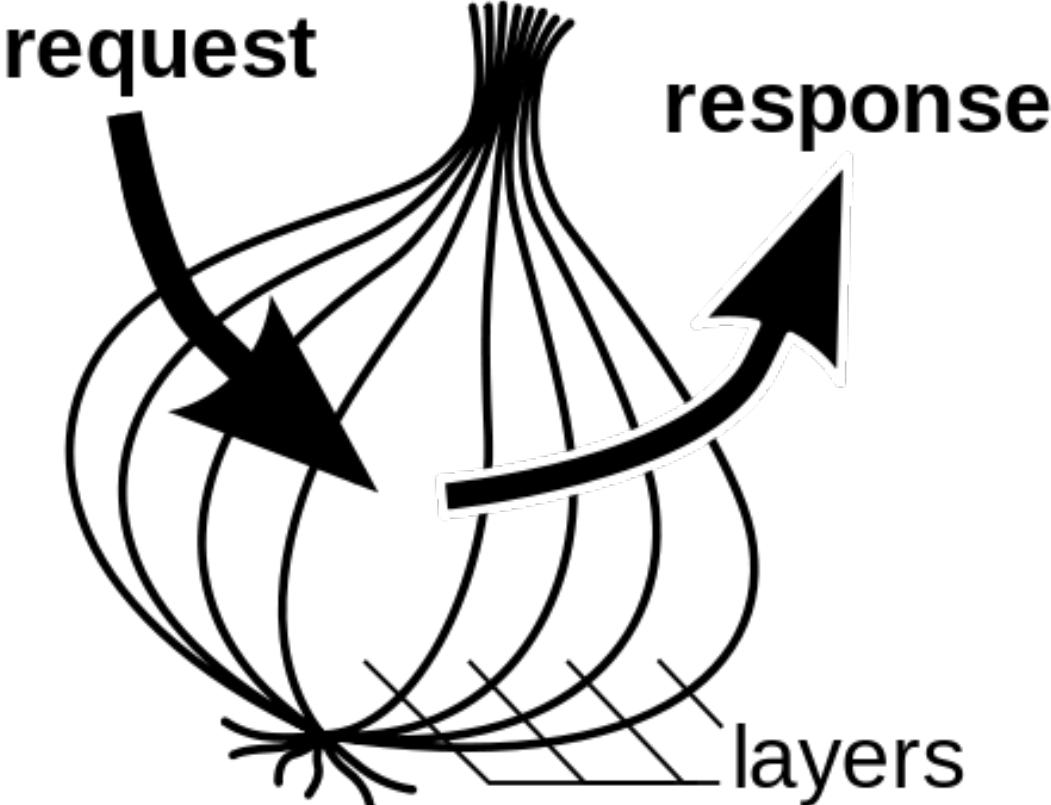
Dancer2::Plugin::Email Sending emails

Dancer2::Plugin::Interchange6 Cart and other business functions

Dancer2::Plugin::TemplateFlute Form handling

Plack Middleware

Joy of PSGI middleware



Cross-site request forgery (CSRF/XSRF)

```
Plack::Middleware::XSRFBlock
```

```
# bin/app.psgi
```

```
use Plack::Builder;
```

```
use PerlDance;
```

```
my $app = PerlDance->to_app;
```

```
builder {
```

```
  enable 'Session';
```

```
  enable 'XSRFBlock',
```

```
    cookie_name => 'PerlDance-XSRF-Token',
```

```
    meta_tag    => 'xsrp-meta',
```

```
    cookie_options => { httponly => 1, };
```

```
  enable 'XForwardedFor',
```

```
    trust => [qw( 127.0.0.1 10.0.0.0/8 172.16.0.0/20 192.168.0.0/16 )];
```

```
  $app;
```

```
}
```

Summary

- Modern Perl
- Missing features
- Perl Dancer Conference 2016
- Call for developers
- Ask questions

Modern Perl

Missing features

- Multilingual GH #8
- Drag and drop schedule tool
- Multisite
- Multiple speakers GH #104

Perl Dancer Conference 2016



Call for developers

<https://github.com/interchange/Perl-Dancer-Conference>

Questions

Questions?

Linuxia Wiki

Stefan Hornburg (Racke)
No Act on ACT
2016-09-21

wiki.linuxia.de