

Welcome to Charlotte Perl Mongers!

Jason A. Crome
Senior Perl Developer, Empowered

Who are we?

What is Perl Mongers?

- Grassroots organization for promoting Perl
- Run by volunteers (and zealots like me!)
- Share and disseminate information about Perl and the community
- Welcome to Charlotte.pm!

Engage with us!

- Us = all of you!
- Use our Meetup
- [charlottedevs.slack.com#perl](https://charlottedevs.slack.com/#perl)
- [irc.perl.org#charlotte.pm](https://irc.perl.org/#charlotte.pm)

What is Modern Perl?



What is Modern Perl?

- First-class OO programming
- Great Testing Infrastructure
- Deployment and Configuration Toolchain
- Active user community
- Modern language features
- Modern frameworks

Who's Using Modern Perl?

- ZipRecruiter
- Booking.com
- Craigslist
- Ticketmaster
- Amazon
- BBC
- NASA
- Rent.com
- DuckDuckGo
- IMDB
- Empowered
- Shutterstock
- ServerCentral
- Pinboard

What is Modern Perl?

- First-class OO programming
- Great Testing Infrastructure
- Deployment and Configuration Toolchain
- Active user community
- Modern language features
- Modern frameworks

Object Oriented

- Moose
- Moo
- Mouse

```
package Point;

sub new {
    my ($class, $x, $y) = @_;
    my $self = {
        x => $x,
        y => $y,
    };
    return bless $self, $class;
}

sub get_x {
    return $_[0]->{x};
}

sub set_x {
    return $_[0]->{x} = $_[1];
}

sub get_y {
    return $_[0]->{y};
}

sub set_y {
    return $_[0]->{y} = $_[1];
}

1;
```

```
package Point;
use Moose;

has 'x' => (isa => 'Int', is => 'rw', required => 1);
has 'y' => (isa => 'Int', is => 'rw', required => 1);

sub clear {
    my $self = shift;
    $self->x(0);
    $self->y(0);
}

package Point3D;
use Moose;

extends 'Point';

has 'z' => (isa => 'Int', is => 'rw', required => 1);

after 'clear' => sub {
    my $self = shift;
    $self->z(0);
};
```

Using the Point Class

```
# hash or hashrefs are ok for the constructor
my $point1 = Point->new(x => 5, y => 7);
my $point2 = Point->new({x => 5, y => 7});

my $point3d = Point3D->new(x => 5, y => 42, z => -5);
```

Roles with Moose

```
package MyApp::CLI::SetPassword;

use strictures 2;
use MooseX::App::Command;
extends qw( MyApp::CLI );
with qw(
    MyApp::Role::CLI::WithPassword
    MyApp::Role::CLI::WithUsername
    MyApp::Role::CLI::WithUser
);
use MyApp::Module;

sub run ( $self ) {
    $self->_say( "Changing password for " . $self->username . '...' );
    $self->user->password( $self->password );
    $self->user->update;
    say "Done!";
}

1;
```

Testing Infrastructure

- `Test2`
- `Test::Class::Moose`
- `prove`
- `TAP`

Deployment/Configuration

- Plack/PSGI
- `cpanfile`
- Carton
- Pinto
- `perlbrew`
- `plenv`

Community

- Perl's greatest strength
- Perl Mongers
- YAPC / TPC
- (Meta)CPAN

MetaCPAN

← → ↻

https://metacpan.org/pod/Dancer2

Aerodrome My Sites Flight Hockey Perl PostgreSQL Javascript/AJAX CSS Vim git Tools

Home grep::cpan Recent About FAQ GitHub Issues News Tools API Sign in

meta::cpan

Search

Jason A. Crome / - Dancer2-0.207000 101 ++ / Dancer2

NOV 14, 2018

Distribution: Dancer2
Module version: 0.207000

[Source \(raw\)](#)
[Browse \(raw\)](#)
[Changes](#)
[Homepage](#)
[How to Contribute](#)
[Repository \(git clone\)](#)
[Issues \(134\)](#)
[Testers \(1135 / 38 / 0\)](#)
[Kwalitee](#)
[84.48% Coverage](#)
[License: perl_5](#)

Questions? Chat with us!

ACTIVITY

TOOLS

PERMALINKS

Contents [hide]

[NAME](#)
[VERSION](#)
[DESCRIPTION](#)
[FUNCTIONS](#)
[SECURITY REPORTS](#)
[SUPPORT](#)
[AUTHORS](#)
[CORE DEVELOPERS](#)
[CORE DEVELOPERS EMERITUS](#)
[CONTRIBUTORS](#)
[AUTHOR](#)
[COPYRIGHT AND LICENSE](#)

NAME
Dancer2 - Lightweight yet powerful web application framework

VERSION
version 0.207000

DESCRIPTION
Dancer2 is the new generation of [Dancer](#), the lightweight web-framework for Perl. Dancer2 is a complete rewrite based on [Moo](#).





Dancer2 can optionally use XS modules for speed, but at its core remains fatpackable (packable by [App::FatPacker](#)) so you could easily deploy Dancer2 applications on hosts that do not support custom CPAN modules.


Dancer2 is easy and fun:


```
use Dancer2;  
get '/' => sub { "Hello World" };  
dance;
```


This is the main module for the Dancer2 distribution. It contains logic for creating a new Dancer2 application.

++ed by:


45 PAUSE users
56 non-PAUSE users.


CROMEDOME
Jason A. Crome
and 1 contributors
[show them](#)

DEPENDENCIES
[App::Cmd::Setup](#)
[Attribute::Handlers](#)
[Carp](#)
[Clone](#)
[Config::Any](#)
[Digest::SHA](#)
[Encode](#)
[Exporter](#)
[Exporter::Tiny](#)
[File::Basename](#)
[File::Copy](#)
[File::Find](#)
[File::Path](#)
[File::Share](#)
[File::Spec](#)
[File::Temp](#)
[Hash::Merge::Simple](#)
[Hash::MultiValue](#)
[HTTP::Date](#)
[HTTP::Headers::Fast](#)
[HTTP::Tiny](#)
[Import::Into](#)
[JSON::MaybeXS](#)

Modern Language Features/Tools

- Named parameters/sub signatures
- Meta-object protocol
- XS
- Pod
- Perlcritic
- Perltidy

Named params/sub signatures

```
sub clear( $self ) {  
    $self->x(0);  
    $self->y(0);  
}
```

```
after 'clear' => sub( $self ) {  
    $self->z(0);  
};
```

```
sub set( $self, $x, $y, $z?) {  
    $self->x( $x );  
    $self->y( $y );  
    $self->z( $z ) if $z;  
}
```

Modern Frameworks

- Object Frameworks
- ORMs
- Templating Frameworks
- Web App Frameworks

ORMs

- `Class::DBI`
- `DBIx::Class`
- `Rose::DB`

DBIx::Class

```
$user = $schema->resultset('User')->find( { email => $self->email } );
```

```
$user = $schema->resultset('User')->create(  
    {  
        username => $self->username,  
        password => $self->password,  
        email    => $self->email,  
    }  
);
```

```
my $user_rs = $schema->resultset('User')->search({  
    email => { like => '%example.com' },  
    }, {  
        order_by => { username => 'ASC' },  
    }  
);
```

Templating Frameworks

- `HTML::Template`
- Mason
- Template Toolkit
- Mustache
- XSlate
- And many, many others...

Web App Frameworks

mojomilicious 

PERLDANXER



And now, live coding
and swag!



Questions?

Thanks!

<https://empowered.net>

jcrome@empoweredbenefits.com

Twitter: @cromedome