

NAME

MooseX::LazyRequire – Required attributes which fail only when trying to use them

VERSION

version 0.11

SYNOPSIS

```
package Foo;

use Moose;
use MooseX::LazyRequire;

has foo => (
    is          => 'ro',
    lazy_required => 1,
);

has bar => (
    is          => 'ro',
    builder => '_build_bar',
);

sub _build_bar { shift->foo }

Foo->new(foo => 42); # succeeds, foo and bar will be 42
Foo->new(bar => 42); # succeeds, bar will be 42
Foo->new;          # fails, neither foo nor bar were given
```

DESCRIPTION

This module adds a `lazy_required` option to Moose attribute declarations.

The reader methods for all attributes with that option will throw an exception unless a value for the attributes was provided earlier by a constructor parameter or through a writer method.

CAVEATS

Prior to Moose 1.9900, roles didn't have an attribute metaclass, so this module can't easily apply its magic to attributes defined in roles. If you want to use `lazy_required` in role attributes, you'll have to apply the attribute trait yourself:

```
has foo => (
    traits      => ['LazyRequire'],
    is          => 'ro',
    lazy_required => 1,
);
```

With Moose 1.9900, you can use this module in roles just the same way you can in classes.

AUTHORS

- Florian Ragwitz <rafl AT debian DOT org>
- Dave Rolsky <autarch AT urth DOT org>

COPYRIGHT AND LICENSE

This software is copyright (c) 2009 by Florian Ragwitz.

This is free software; you can redistribute it and/or modify it under the same terms as the Perl 5 programming language system itself.

CONTRIBUTORS

- Karen Etheridge <ether AT cpan DOT org>
- David Precious <davidp AT preshweb DOT co DOT uk>
- Jesse Luehrs <doy AT tozt DOT net>

