NAME

MooseX::Storage::Traits::DisableCycleDetection – A custom trait to bypass cycle detection

SYNOPSIS

```
package Double;
use Moose;
use MooseX::Storage;
with Storage( traits => ['DisableCycleDetection'] );

has 'x' => ( is => 'rw', isa => 'HashRef' );
has 'y' => ( is => 'rw', isa => 'HashRef' );

my $ref = {};

my $double = Double->new( 'x' => $ref, 'y' => $ref );

$double->pack;
```

DESCRIPTION

MooseX::Storage implements a primitive check for circular references. This check also triggers on simple cases as shown in the Synopsis. Providing the DisableCycleDetection traits disables checks for any cyclical references, so if you know what you are doing, you can bypass this check.

This trait is applied to all objects that inherit from it. To use this on a per-case basis, see disable_cycle_check in MooseX::Storage::Basic.

See the SYNOPSIS for a nice example that can be easily cargo-culted.

METHODS

Introspection

meta

BUGS

All complex software has bugs lurking in it, and this module is no exception. If you find a bug please either email me, or add the bug to cpan-RT.

AUTHOR

Stevan Little <stevan DOT little AT iinteractive DOT com>

COPYRIGHT AND LICENSE

Copyright 2007−2008 by Infinity Interactive, Inc.

<http://www.iinteractive.com>

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