NAME

MooseX::OneArgNew – teach –>new to accept single, non–hashref arguments

VERSION

version 0.002

SYNOPSIS

In our class definition:

```
package Delivery;
use Moose;
with('MooseX::OneArgNew' => {
  type    => 'Existing::Message::Type',
  init_arg => 'message',
});

has message => (isa => 'Existing::Message::Type', required => 1);

has to => (
  is   => 'ro',
  isa  => 'Str',
  lazy => 1,
  default => sub {
    my ($self) = @_;
    $self->message->get('To');
  },
);
```

When making a message:

```
# The traditional way:

my $delivery = Delivery->new({ message => $message });
# or
my $delivery = Delivery->new({ message => $message, to => $to });

# With one-arg new:

my $delivery = Delivery->new($message);
```

DESCRIPTION

MooseX::OneArgNew lets your constructor take a single argument, which will be translated into the value for a one-entry hashref. It is a parameterized role with two parameters:

type

The Moose type that the single argument must be for the one-arg form to work. This should be an existing type, and may be either a string type or a MooseX::Type.

init_arg

This is the string that will be used as the key for the hashref constructed from the one-arg call to new.

WARNINGS

You can apply MooseX::OneArgNew more than once, but if more than one application's type matches a single argument to new, the behavior is undefined and likely to cause bugs.

It would be a **very bad idea** to supply a type that could accept a normal hashref of arguments to new.

AUTHOR

Ricardo Signes <rjbs AT cpan DOT org>

COPYRIGHT AND LICENSE

This software is copyright (c) 2011 by Ricardo Signes.

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