

failures

exceptions

- Try::Tiny is fairly awesome
- Regex to figure out the exception type isn't

Using it

```
use failures qw/broker/;
use Try::Tiny;
use Safe::Isa;

try
{
    failure::broker->throw('testing');
}
catch
{
    if($_->$_isa('failure::broker'))
    {
        print $_;
        print "Message ", $_->msg, "\n"; # big win
    }
};
```

What's the big deal?

It allows you to separate,

- where the exception came from
- the specific error message - e.g. you can make it human readable

Safe::Isa

Mentioned in MST's lightning talk at LPW2013

A safe way to check do 'isa' on something that may or may not be an object. Saves a bunch of 'is this blessed' etc. checks.

What about our existing code?

The existing exception can work as is, you can simply enhance your throwing and catching where necessary. Spotting 'failure' classes and strings can work together. Especially easy thanks to `Safe::Isa`

Features to note

- Stack traces are optional - you need to explicitly provide them when throwing if you want them
- If you want to provide extra information along with the exception you can pass it in the payload attribute

In case you were worried

- It's a DAGOLDEN module (an author I trust)
- It has minimal dependencies