

Currying Logs

This article is a tangent on Currying Databases.

A little background: Our student portal was riddled with bugs. Since we use a monolithic system like ASP.Net Classic, any tiny issue could cause our entire page to crash. The codebase is riddled with sloppy code handed down from person to person through 5 generations of programmers who formerly held the position. These people had no interest in cleaning up the code or making it coherent. To make matters worse, the vendor themselves had bugs that they did not fix.

The following I dealt with these issues.

1. Break giant operations into smaller ones. Test it, make sure it works after you've refactored it.
2. Wrap these smaller operations in a Try-Catch.
3. When there is a chance for failure, catch it and log it and handle the error scenario gracefully to the user.
4. When logging the information, supply as much information as you can. Provide a concise description of the error so it will jot your memory when it happens.
5. Try to anticipate errors. Keep an eye on the log, create different categories of alerts for the errors you get. This way you may be able to see errors as your users interact with your application.
6. Create a service that will aggregate the errors tripped in the last 24 hours. Then fix them.

Currying is a great technique can can be used for logging.

How?

Currying allows us to string together expressions and execute them.

This is particularly useful when we need to supply important information inputs into a function.

```
const string cmdTemplate = "select * from _form_templates where id = @id  
and category = 'Course Forms (Pre-offer)' ";  
  
try  
{  
    model = Db  
        .Forms  
        .Cmd(cmdTemplate)  
        .Param("@id", info.formid)  
        .Row();  
}  
catch (Exception ex)  
{  
    ex.Error("[PZ Courses Edit] Error when trying to select the form  
templates.")  
    .Add("cmdTemplate", cmdTemplate)  
    .Add("model", model.Json())
```

```
        .Add("input", info.Json())  
        .Ok();  
  
    return ex.Handle();  
}
```