Ruby - Feature #18659

Create a Binding at the time of an exception and make it available to Rescue

03/25/2022 02:02 AM - pedz (Perry Smith)

Status:	Open	
Priority:	Normal	
Assignee:		
Target version:		

Description

When an exception occurs, create a binding or I assume make a reference to the current binding and then add that to the Exception class via (e.g.) a #binding method. This would allow users to have a more global rescue code but would still have access to the variables that were in scope at the time of the exception.

My particular use case is when parsing files -- especially a list of files. The outer loop is for each file. The next inner loop is for each line in the file. At the time of an exception, it would be nice to have access while in the a rescue clause to the file name, line number, current line, and other interesting variables. Adding rescue closes in the inner blocks creates a lot of redundant code.

History

#1 - 03/25/2022 10:55 AM - Eregon (Benoit Daloze)

This would be a huge overhead, so I don't think it's a good idea.
If you need this data, you could just pass it down yourself (e.g., the inner loop could be a block and then you already have access to all variables, or you could pass data like the file as arguments), and that way wouldn't slow down ever other exception.