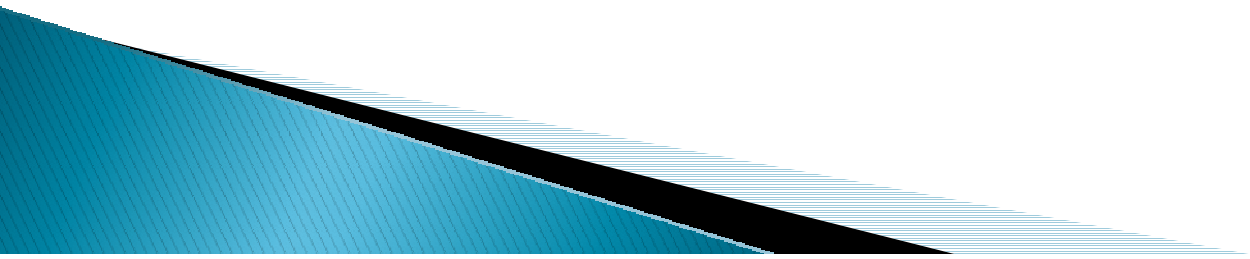
errorhandling techniques in XPages, Javascript, Java, Lotusscript and @Formulas

Agenda

- What is error handling?
- Errorhandling in:
  - @Formula
  - Javascript
  - Lotusscript
  - Java
  - xPages
- Openlog

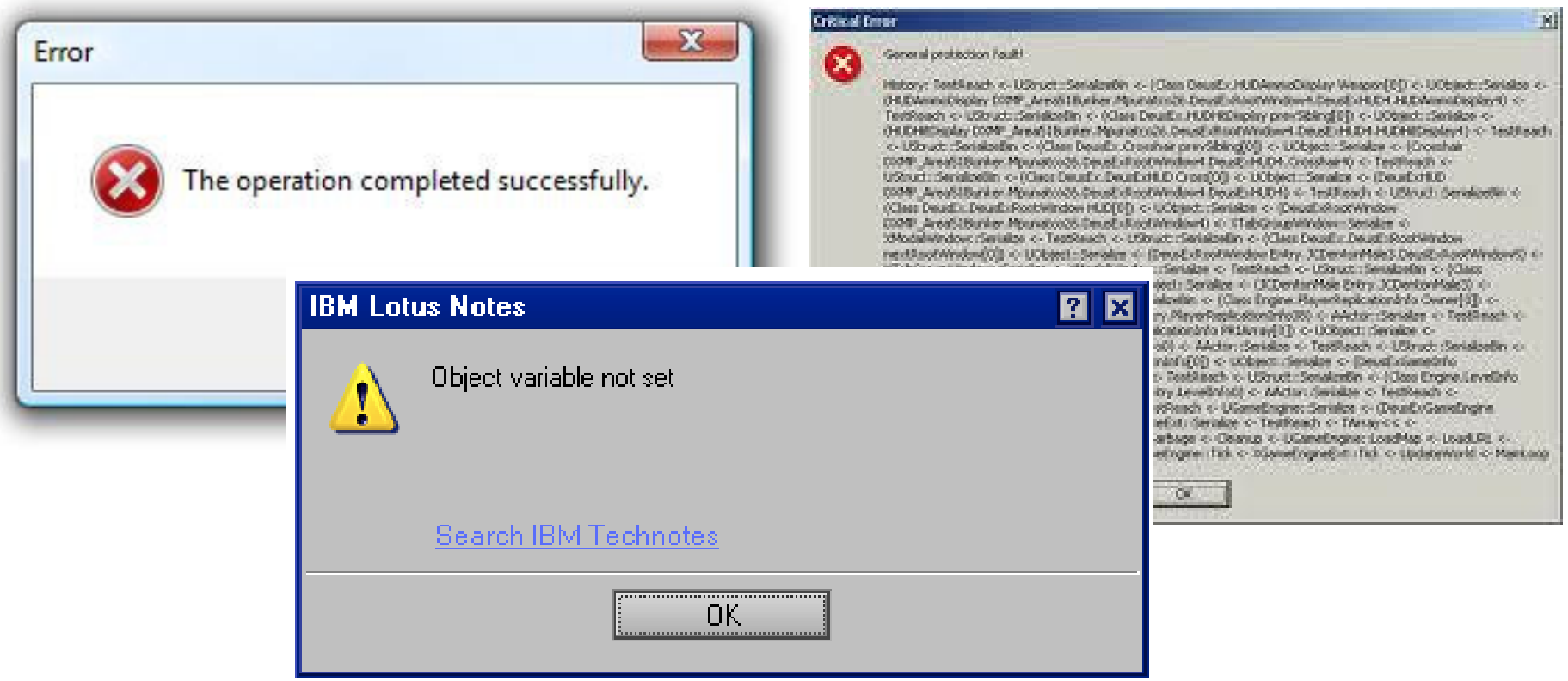


Error handling

Anticipate ser and environme

Prevent p

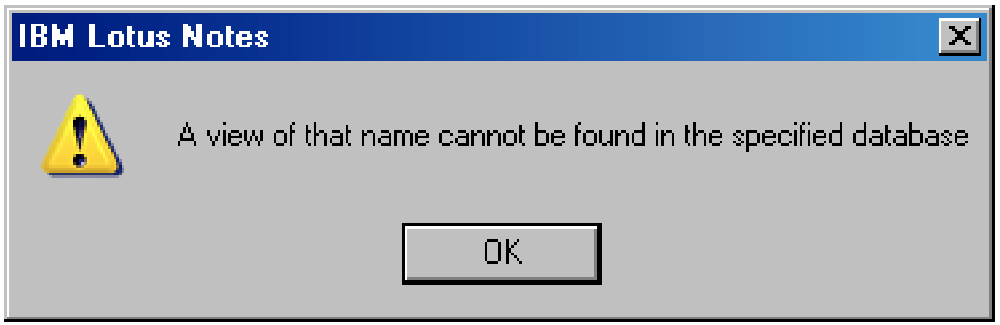
What cou


Error
The operation completed successfully.
Critical Error
General protection fault
IBM Lotus Notes
Object variable not set
Search IBM Technotes
OK

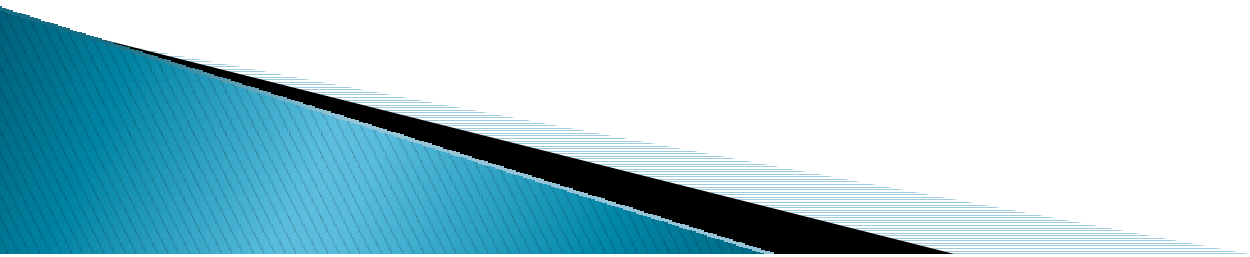
@Formulas

Formula Language

- Loads of cryptical error messages – no info about triggered events
- No central error handling possible - be sparse with your @Formulas - use display fields for re-usable calculations
- Defensive programming (@IsError, Test DataTypes: @Elements, @IsTime, @IsNumber, @Text ...)



Javascript



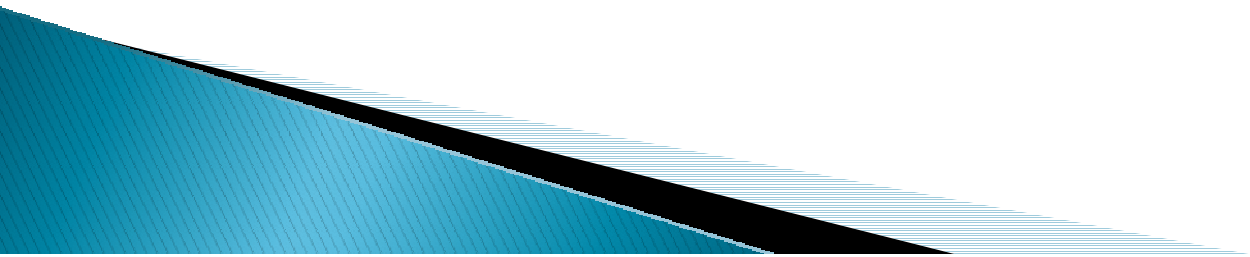
Javascript

Standard try - catch routine

```
try {
    // call a function that could error out
    var result = getInvoiceNumbers(customerNumber,invoiceDate);;
}
catch (e) {
     result = "Error in retrieving invoices";
}
// optional use finally runs after the try/catch block
finally {
     window.Forms[0].fldname.value = result
}
```

Throw your own errors

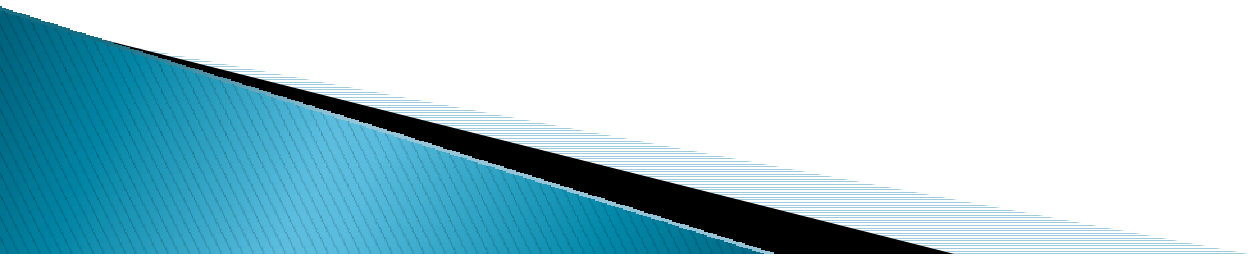
```
if (number > 10) { throw "NumberTooHighError" }
```



Javascript

Try - catch routine based on error type

```
try {
    var result = getInvoiceNumbers(customerNumber,invoiceDate);
}
catch (e
    if e == "CustomerNumberException") {
        result = "Incorrect customerNumber";
    }
   else if e == "DateErrorException") {
        result = "Incorrect Date";
    }
   else result = "Error in retrieving invoices";
}
```



Javascript

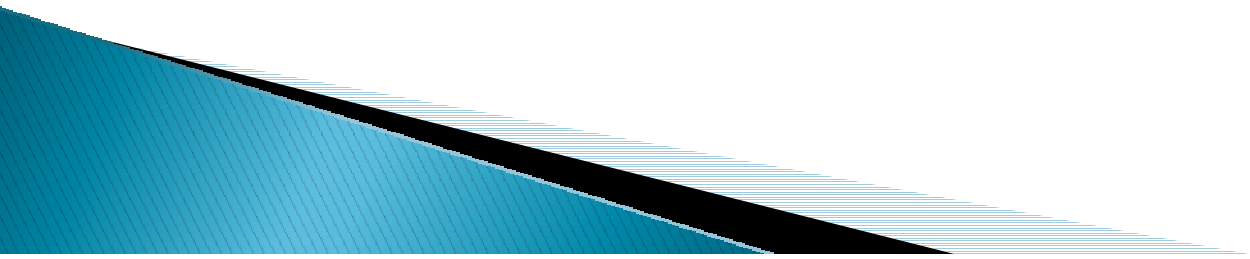
To register a general errorHandler use:

```
(window.)onerror=functionname()
```

This function will have 3 input variables like:

```
function functionname(msg,url,line_num)
```

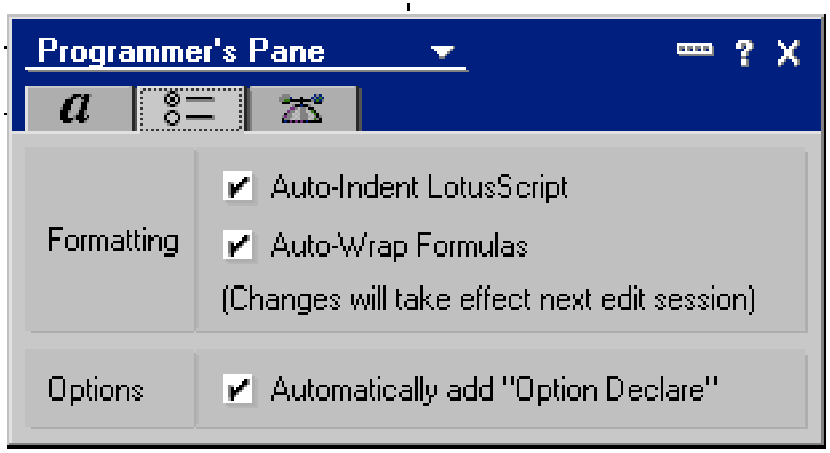
You can use these variables to log/show the error details



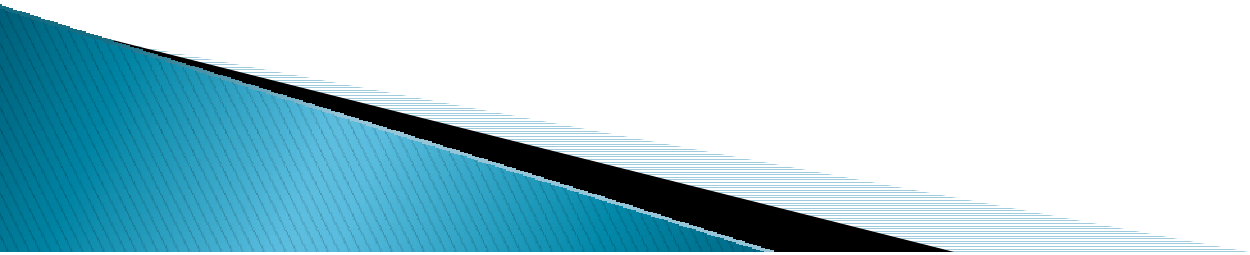
Lotusscript

Lotusscript

- use `On Error` blocks
- `On Error resume nex` upposed to handle things, nc
- You have differte opers, administrators. S
- Then handle spe
- `Option Declare` pre errors
- *new!* Create a template in the new 8.5.1 LS IDE for your error handling blocks in subs and functions



Java



Java

- As with JavaScript use try - catch - finally blocks and throw/catch custom errors

```
Try {
    List list = new ArrayList(getInvoiceNumbers(customerNumber,invoiceD
    ate));
} catch ( IOException e) {
     // handle/log this error
} catch ( OtherErrorException e) {
     // again handle this
} catch (Exception e) {
     // other errors
}
// optional
}finally {
}
```

Java

- Chose:
  - Handle errors in subfunctions
  - Throw the error back to higher functions

```
private Integer convertToInteger(String str) {
try {
    Integer intObj2 = Integer.valueOf(str)
    } catch (e) { // do something }
    return int;
}

private Integer convertToInteger(String str) throws
NumberFormatException {
    Integer intObj2 = Integer.valueOf(str)
}
```

Java

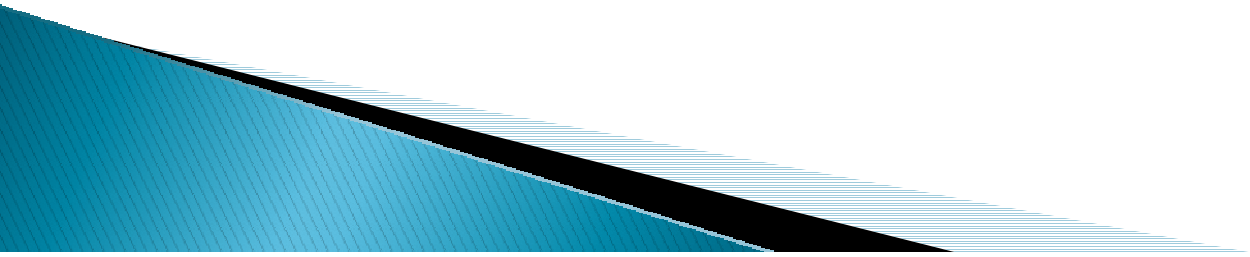
```
class CustomException extends Exception {
    public CustomException(String msg){
            super(msg);
       }
}

public class Test {

 public static void main(String[] args) {
  try {
    System.out.println(divide(4,0));
  }
    catch (CustomException e) { e.printStackTrace(); }
  }

static int  divide(int first,int second) throws CustomException{
  if(second==0)  throw new MyException("dividing by zero");
  return first/second;
 }
}
```

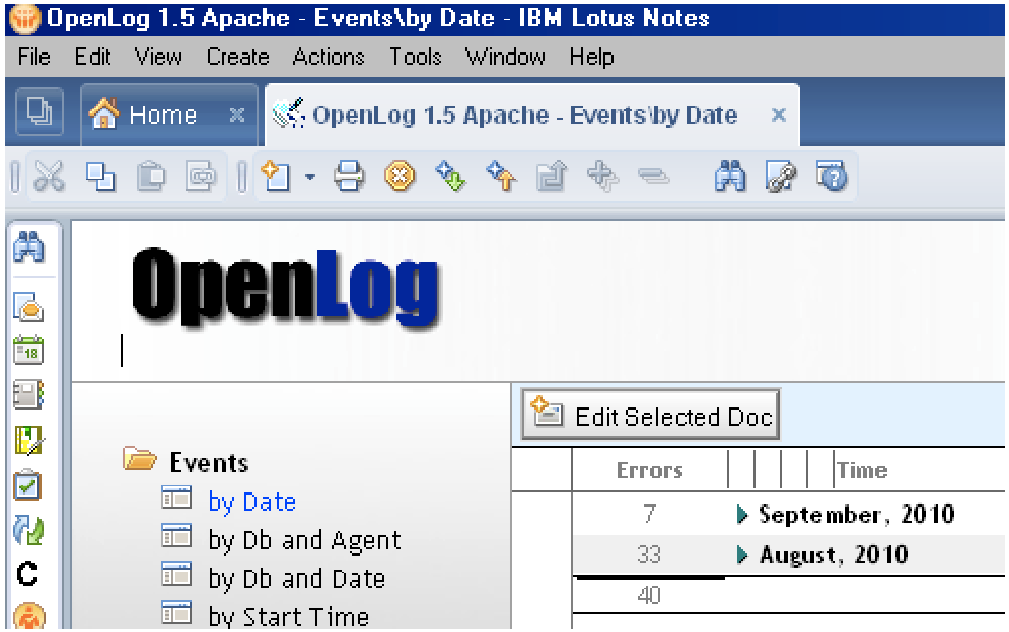
xPages

xPages

Error handling in xPages

- Default errorpage on server (admin can change xsp\nsf\xsp.properties)
- Set errorpage in application
- Custom Error page (general, debug, specific)
- Logging to AgentLog





Openlog

how wonderful, free stuff

- JavaScript
- Java
- XPages
- Lotusscript

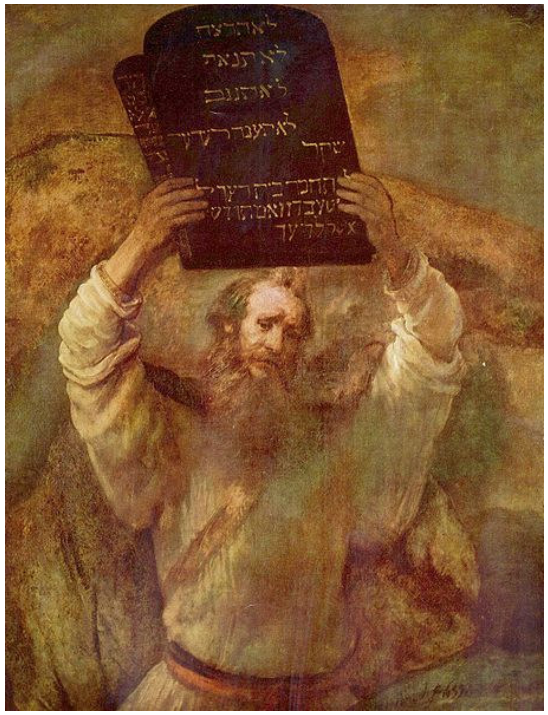
DEMO!!!!!!

Ten Commandments

- Getting it to work is NOT ENOUGH!
- Always add errorhandling to production code
- Thou shall avoid the use of hard coded stuff
- When not sure: check
- If it can go wrong, it will go wrong. At one point.
- Think first, then build your logic
- Do not repeat code
- Use only one line of code in events (click, query*)
- Your users are dumb ...

but: you are even dumber (if you ignore them)

- Make your code beautiful: comment not what, but why and use meaningfull names



Contact us

Vince Schuurman
vince.schuurman@gmail.com

Martin Schaefer
martin@mschaefer.nl