

Incremental Return Codes.

Be Warned! About Everything.

Van Krunkelsven Dirk



Two types of trouble in a program

```
0101101010001001000001111010101010011101101010100100111001010111101001001
100111011010101010010011100101011110100100111001101001011110101101010010100
0111001001001001001110110101010010011100101011110100100111001101001010011
1001110010100101001101010100100111001010111101001001110011010010111101011
0101101010001001000001111010101010011101101010100100111001010111101001001
100111011010101010010011100101011110100100111001101001011110101101010010100
0111001001001001001110110101010010011100101011110100100111001101001011110
1001110010100101001101010100100111001010111101001001110011010010111101011
0101101010001001000001111010101010011101101010100100111001010111101001001
100111011010101010010011100101011110100100111001101001011110101101010010100
101000001110010010010010011101101010100100111001010111101001001110011010
1001110010100101001101010100100111001010111101001001110011010010111101011
0101101010001001000001111010101010011101101010100100111001010111101001001
100111011010101010010011100101011110100100111001101001011110101101010010100
0111001001001001001110110101010010011100101011110100100111001101001011110
1001110010100101001101010100100111001010111101001001110011010010111101011
0101101010001001000001111010101010011101101010100100111001010111101001001
100111011010101010010011100101011110100100111001101001011110101101010010100
0111001001001001001110110101010010011100101011110100100111001101001011110
1001110010100101001101010100100111001010111101001001110011010010111101011
010110101000100100000111101010101001110110101010010011100101011110100100
100111011010101010010011100101011110100100111001101001011110101101010010100
101000001110010010010010011101101010100100111001010111101001001110011010
1001110010100101001101010100100111001010111101001001110011010010111101011
```

1. It's Broken

- It's broken or it will break apart
 - Makes good sense to STOP





2. Something not 100%

- Can CONTINUE but...
- BE WARNED



**BEWARE
UNFENCED ROAD**



FOR NEXT 150km

Keep Track of Multiple Issues

- Issue encountered
- Program continues
- Another issue encountered
- Multiple issues possible
- Keep track of all to report back to the user

Option: Multiple Macro Variables

Each in an ON/OFF state

- Quickly too much to manage
 - Control existence
 - Interpret each
- Confusing





GOOD LUCK

45
M.P.H.

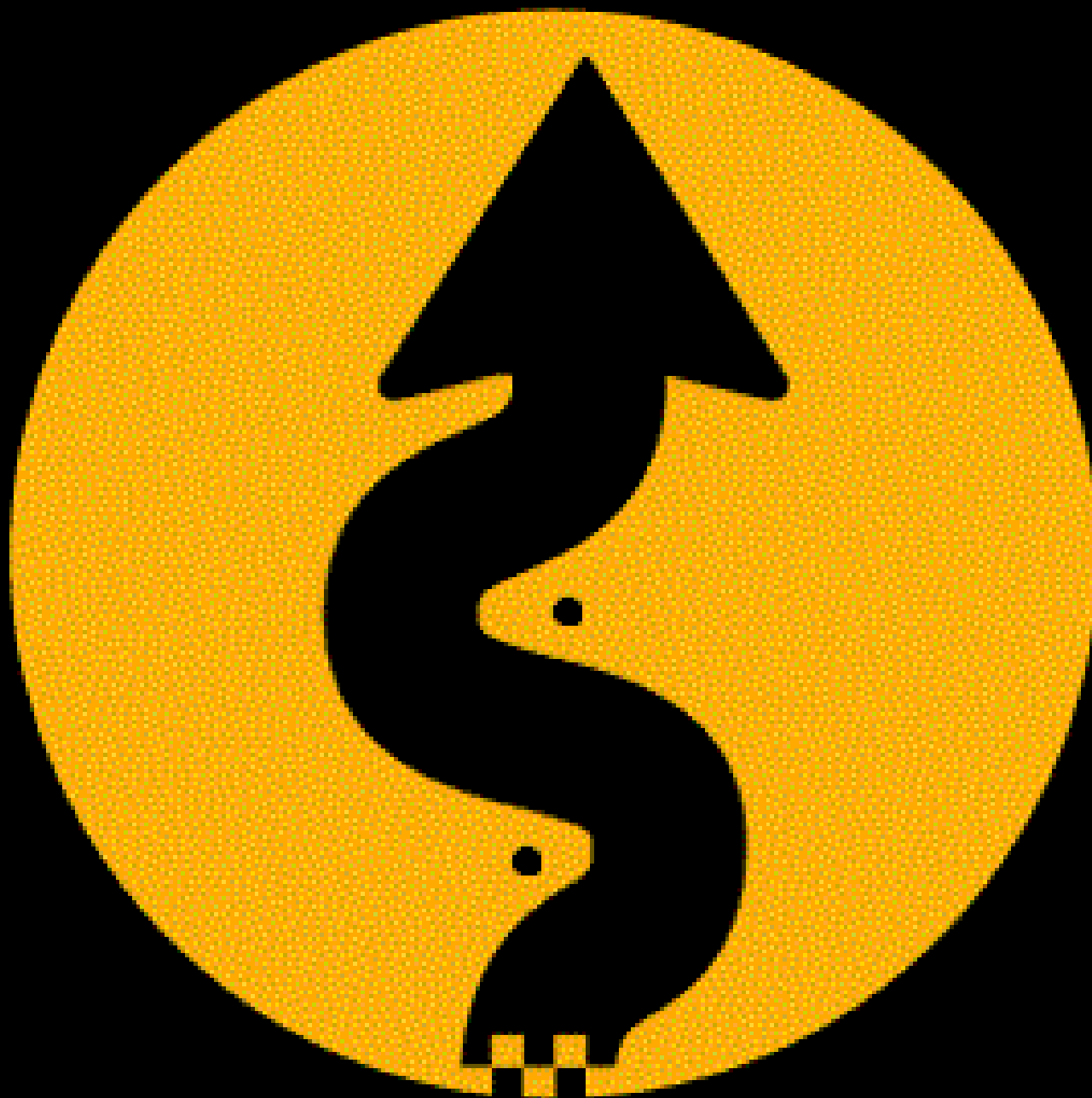


Option: One Incremental Macro Var.

A single incremental macro variable

- Increments for each issue
- Easy to manage
- Easy to interpret
- More straightforward





Incremental Return Codes

Values: 1 2 4 8 16 32 64 128 256 512 1028 ...

- Powers of 2
- Binary representation: only one bit is ON
 - 0000 0001 (1) – Event A
 - 0000 0010 (2) – Event B
 - 0000 0100 (4) – Event C
 - 0000 1000 (8) – Event D
 - 0001 0000 (16) – Event E
 - ...

Combining: Bitwise OR

- Combining values: bitwise OR
 - $\text{bOR}(0010, 0001) = 0011$

	0	0	0	1	1
BOR	0	0	1	0	2
	0	0	1	1	3

Unravel to components (events)

- Easy to go back to the components
 - $0101 = \text{bOR}(0100, 0001)$
 - $5 = \text{bOR}(4, 1)$

0000 0001	(1) – Event A
0000 0010	(2) – Event B
0000 0100	(4) – Event C
0000 1000	(8) – Event D
0001 0000	(16) – Event E

- Events A and C occurred

Combining: Bitwise OR

- Combining same event: value does not change

	0	0	1	0	2
BOR	0	0	1	0	2
	0	0	1	0	2

Implementation: Supp2par_v1.sas

[www.code.google.com/p/phuse-scripts/
downloads/list](http://www.code.google.com/p/phuse-scripts/downloads/list)

SAS Code

■ Setting the Return Code

```
%let RC_WAR=%sysfunc(bOR(&RC_WAR,128)) ;
```

■ Interpreting the Return Code

```
if RC = '1'b          then put 'WARN' 'ING- * Event 1 occurred';  
if RC = '1.'b         then put 'WARN' 'ING- * Event 2 occurred';  
if RC = '1..'b        then put 'WARN' 'ING- * Event 3 occurred';  
if RC = '1...'b       then put 'WARN' 'ING- * Event 4 occurred';  
if RC = '1....'b      then put 'WARN' 'ING- * Event 5 occurred';
```

Thank you...

