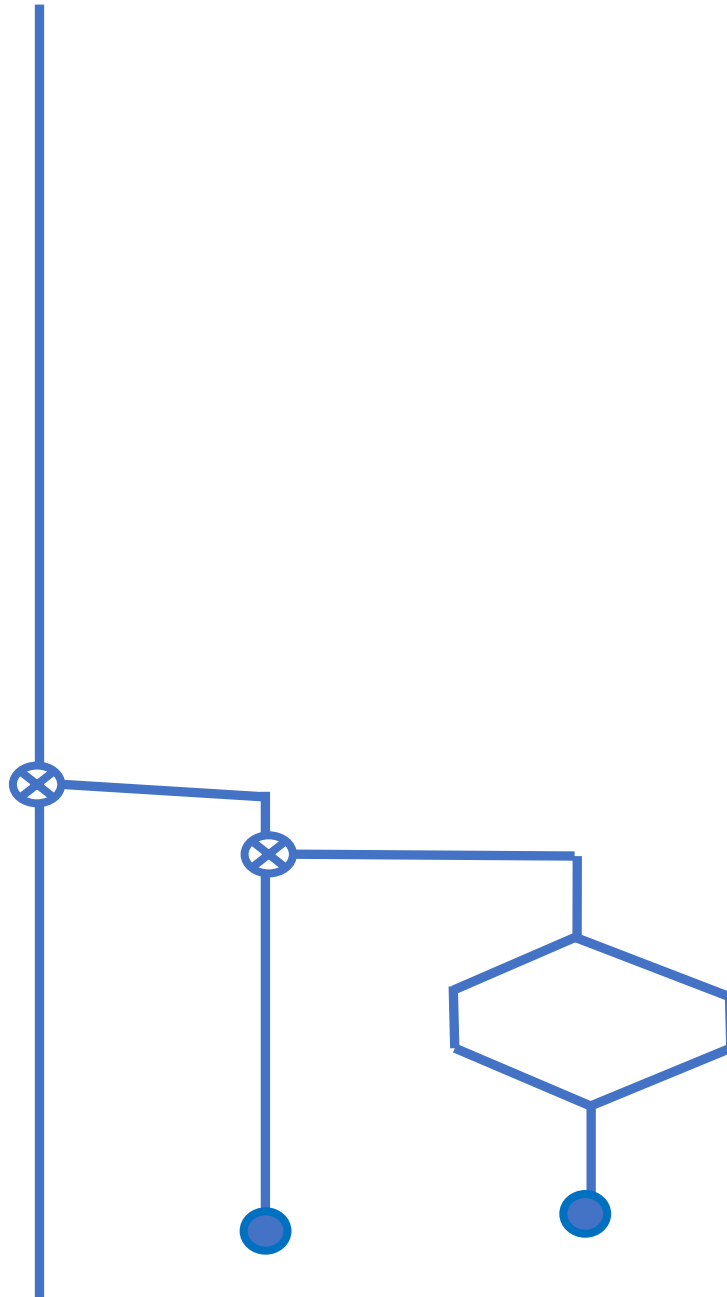


# Maximizing Your Forth Presentation

Silicon Valley Forth  
Interest Group  
Nov. 20, 2021  
Bill Ragsdale



# Today

For our world-wide Forth2020, Zoom is our platform and our medium.

# Today

For our world-wide Forth2020, Zoom is our platform and our medium.

Increasingly it has replaced print as our archive.

# Today

For our world-wide Forth2020, Zoom is our platform and our medium.

Increasingly it has replaced print as our archive.

Let's make the most of that resource.

# Why

Let's improve the clarity and utility of our monthly presentations.



# Why

Let's improve the clarity and utility of our monthly presentations.

We are doing a deep dive into technical topics to a diverse audience.



# Why

Let's improve the clarity and utility of our monthly presentations.

We are doing a deep dive into technical topics to a diverse audience.

And time flies.



# Contra-Factual

You can't talk for twenty minutes, scrolling a screen of unfamiliar code and hold the viewer's attention.

```
73      IF c" open slot"      ENDIF \ open
74      drop c" found no match"  ENDORSE ;
75
76 \ *** Experimental error code array *** \
77
78 [ constant error-cells \ eight plus one for the pointer
79
80 create Error-Array error-cells 1+ cells allot \ first cell is pointer
81 \ space for up to 6 sequential set-check values
82
83 create Name-Array error-cells 1+ cells allot \ cfa of error calling word
84
85 : reset-error-array
86   Error-Array error-cells 1+ cells erase
87   Name-Array error-cells 1+ cells erase
88   ['] noop Name-Array cell+ ! ;
89
90 reset-error-array
91
92 : reduce-error-code ( --- ) \ at end of a code sequence
93 \ only reduce error code if > 0
94   error-array @
95   dup 0 > swap error-cells < and
96   if -1 error-array +! then ;
97
98 : last-error-code \ address of last error code
99   error-array dup @ cells+ ;
100
101 : last-error-name \ address of last error word cfa
102   Name-Array Error-Array @ cells+ ;
103
104 : set-check ( n --- ) \ add to Error-Array
105   error-array @ error-cells < \ not at end?
106   if 1 error-array +!
107     last-error-code !
108     r@ Last-Error-Name ! else drop then ;
109
110 (( Testing set-check and reduce-error-code
111 : xxx cr error-array 24 dump cr ;
112 reset-error-array xxx
113 [ set-check xxx last-error-code @ .
114 [ set-check xxx last-error-code @ .
115 [ set-check xxx last-error-code @ .
116 reduce-error-code xxx last-error-code @ .
117 reduce-error-code xxx last-error-code @ .
118 reduce-error-code xxx last-error-code @ .
119 reduce-error-code xxx last-error-code @ . ))
120
121
122 sys-warning-off
123 : error-report ( addr cfa1 cfa2 )
124 \ addr is the address of the counted string, error message
125 \ is the cfa1 of the second above calling word
126 \ is the cfa2 of the calling word.
127 [ cr cr last-error-code @ error-table count type space \ x{ y{
128   last-error-name @ ?.name space \ calling word }list
129 [ rot count type space \ message
130   .. in " ??.name space \ a level higher
131   .. by " ??.name ; sys-warning-on
132
133 sys-warning-off
134 : error-report ( addr cfa1 cfa2 )
135 \ addr is the address of the counted string, error message
136 \ is the cfa1 of the second above calling word
137 \ is the cfa2 of the calling word.
138 [ cr cr error-code @ error-table count type space \ x{ y{
139   error-name @ ?.name space \ calling word }list
140 [ rot count type space \ message
141   .. in " ??.name space \ a level higher
142   .. by " ??.name ; sys-warning-on \ error calling word.
143
144 sys-warning-off
145 : {??} ( f -- ) \ gives error report for non-zero flag
146 {(')} ( flag warning-addr ) SMP { warning-addr flag )
147   if r> p> \ warning-addr cfa1 cfa2
148     error-report ( reset-error-code ) reset-error-array
149     cr -2 throw ( to show source text ) THEN DROP ( no error ) ;
150
151 : ?? { -- } \ Error report used in place of abort".
```



# Contra-Factual

You can't talk for twenty minutes, scrolling a screen of unfamiliar code and hold the viewer's attention.

Only if you are Chuck Moore; so don't try.

```
73      OF c" open slot"      ENDOF \ open
74      drop c" found no match" ENDCRSE ;
75
76 \ *** Experimental error code array *** \
77
78 [ constant error-cells \ eight plus one for the pointer
79
80 create Error-Array error-cells 1+ cells allot \ first cell is pointer
81 \ space for up to 6 sequential set-check values
82
83 create Name-Array error-cells 1+ cells allot \ cfa of error calling word
84
85 : reset-error-array
86   Error-Array error-cells 1+ cells erase
87   Name-Array error-cells 1+ cells erase
88   [ ' ] noop Name-Array cell+ ! ;
89
90 reset-error-array
91
92 : reduce-error-code ( --- ) \ at end of a code sequence
93 \ only reduce error code if > 0
94   error-array @
95   dup 0 > swap error-cells < and
96   if -1 error-array +! then ;
97
98 : last-error-code \ address of last error code
99   error-array dup @ cells+ ;
100
101 : last-error-name \ address of last error word cfa
102   Name-Array Error-Array @ cells+ ;
103
104 : set-check ( n --- ) \ add to Error-Array
105   error-array @ error-cells < \ not at end?
106   if 1 error-array +!
107     last-error-code !
108     r@ Last-Error-Name ! else drop then ;
109
110 (( Testing set-check and reduce-error-code
111 : xxx cr error-array 24 dump cr ;
112 reset-error-array xxx
113 [ set-check xxx last-error-code @ .
114 [ set-check xxx last-error-code @ .
115 [ set-check xxx last-error-code @ .
116 reduce-error-code xxx last-error-code @ .
117 reduce-error-code xxx last-error-code @ .
118 reduce-error-code xxx last-error-code @ .
119 reduce-error-code xxx last-error-code @ . ))
120
121
122 sys-warning-off
123 : error-report ( addr cfa1 cfa2 )
124 \ addr is the address of the counted string, error message
125 \ is the cfa1 of the second above calling word
126 \ is the cfa2 of the calling word.
127 [ cr cr last-error-code @ error-table count type space \ x{ y{
128   last-error-name @ ?.name space \ calling word }list
129 [ rot count type space \ message
130   .. in " ??.name space \ a level higher
131   .. by " ??.name ; sys-warning-on
132
133 sys-warning-off
134 : error-report ( addr cfa1 cfa2 )
135 \ addr is the address of the counted string, error message
136 \ is the cfa1 of the second above calling word
137 \ is the cfa2 of the calling word.
138 [ cr cr error-code @ error-table count type space \ x{ y{
139   error-name @ ?.name space \ calling word }list
140 [ rot count type space \ message
141   .. in " ??.name space \ a level higher
142   .. by " ??.name ; sys-warning-on \ error calling word.
143
144 sys-warning-off
145 : {??} ( f -- ) \ gives error report for non-zero flag
146 {(')} ( flag warning-addr ) SMP ( warning-addr flag )
147 if r> p> \ warning-addr cfa1 cfa2
148   error-report ( reset-error-code ) reset-error-array
149   cr -2 throw ( to show source text ) THEN DROP ( no error ) ;
150
151 : ??' ( -- ) \ Error report used in place of abort'.
```



# Some Tools, overview

Use a proven structure.

Use effective communication methods.

Break the overall into smaller bytes (pun intended).

Maximize screen clarity; camera and text.

# First Structure

The simplest protocol.

# First Structure

The simplest protocol.

- 1) Tell 'em what you going to tell 'em.
- 2) Tell 'em,
- 3) Tell 'em what you told 'em.

# First Structure

The simplest protocol.

- 1) Tell 'em what you going to tell 'em.
- 2) Tell 'em,
- 3) Tell 'em what you told 'em.

Or: Beginning, Middle and End.

# First Structure

The simplest protocol.

- 1) Tell 'em what you going to tell 'em.
- 2) Tell 'em,
- 3) Tell 'em what you told 'em.

Or: Beginning, Middle and End.

Or: Introduction, The Topic, Summary.

# Second Structure

Another another model: **SAFW**.

1) **S**tate your idea.

# Second Structure

Another another model: **SAFW**.

- 1) **S**tate your idea.
- 2) **A**ccentuate your idea or its details.



# Second Structure

Another another model: **SAFW**.

- 1) **S**tate your idea.
- 2) **A**ccentuate your idea or its details.
- 3) **F**or instance your idea.

# Second Structure

Another another model: **SAFW**.

- 1) **S**tate your idea.
- 2) **A**ccentuate your idea or its details.
- 3) **F**or instance your idea.
- 4) **W**indup with your idea.

# Over-arching Factor

The most important concept from today.

# Over-arching Factor

The most important concept from today.

Even if you discard everything else.

Before you present.

State to yourself . . . .

# Over-arching Factor

**WHAT IS YOUR INTENDED OUTCOME?**

# YOUR INTENDED OUTCOME

Transfer a skill.

Expand some details for the experienced.

Announce a discovery.

A pro and con argument.

Overview of a topic.

# YOUR INTENDED OUTCOME

Transfer a skill.

Expand some details for the experienced.

Announce a discovery.

A pro and con argument.

Overview of a topic. <<TODAY

# Some Mechanics

The information paths to the brain are visual and auditory.

BUT . . . Visual means **IMAGES** NOT text.

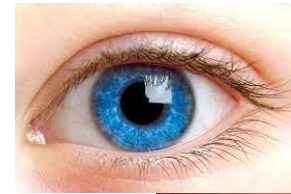


# Some Mechanics

The information paths to the brain are visual and auditory.

BUT . . . Visual means **IMAGES** NOT text.

Visual text and hearing share the same path.



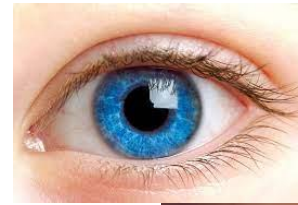
# Some Mechanics

The information paths to the brain are visual and auditory.

BUT . . . Visual means **IMAGES** NOT text.

Visual text and hearing share the same path.

And reading will over-ride hearing.



# The Implication

If you show a slide with four bullet items, your listener/viewer will read ahead while you are talking and miss your comments on point Number One.

# The Implication

If you show a slide with four bullet items, your listener/viewer will read ahead while you are talking and miss your comments on point Number One.

Lemma 1: “Visual leads listening.”

# The Implication

If you show a slide with four bullet items, your listener/viewer will read ahead while you are talking and miss your comments on point Number One.

Lemma 1: “Visual leads listening.”

Lemma 2: “Reveal text only as needed.”

# The Implication

If you show a slide with four bullet items, your listener/viewer will read ahead while you are talking and miss your comments on point Number One.

Lemma 1: “Visual leads listening.”

Lemma 2: “Reveal text only as needed.

Lemma 3: “Or reveal all the bullet points and pause for a few seconds.”

# How To Hold Your Viewer's Interest

Unless you are in an extended workshop format . . . .

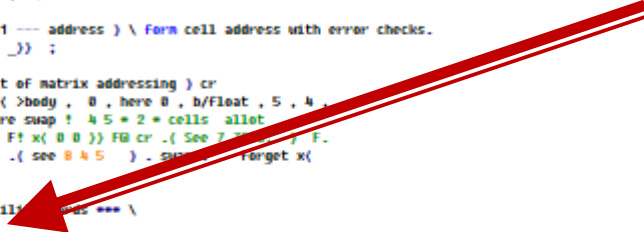
No more than three points.

No more than 15-20 minutes. TED Talks are 20 minutes for a reason .

# Window Sizing

```
317 Forget x( [THEN]
318
319 \ *** Addressing matrix cells *** \
320 ((
321 Use )) until a word has been fully tested.
322 IF parameters are validated at the beginning of a word then
323 replace )) with ())) which is faster, but without error checks.
324 ))
325
326 : )) ( x( r1 c1 --- addr ) \ form a matrix cell address
327   swap 2pick )cols \ x( c r E
328     * \ x( c r+E
329     + over \ x( c+(r+E) x(
330     size: 0 = \ x( (c+(r+E)*size
331     * ; \ addr
332
333 : )) ( x( r1 c1 --- address ) \ form cell address with error checks.
334   ?x(r1c1 _)) ;
335
336 *IF .{ 2 Test of matrix addressing } cr
337 value x( ' x( >body , 0 , here 0 , b/Float , 5 , 4 ,
338 here to x( here swap ! 4 5 = 2 = cells allot
339 x( 0 0 )) 7.7e F! x( 0 0 )) F! cr .( See 7.7e F.
340 cr x( << rot .( see 8 4 5 ) . swap forget x(
341 [then]
342
343 \ *** Matrix utilities *** \
344
345 : )list ( x( --- ) \ list contents, direct with few dependencies.
346   ?x( dup )dimensions swap cr \ a( C R
347   do ( by row ) dup 0 do ( by column )
348     over j 1 _)) F! F.
349   loop cr loop 2drop reduce-error-code ;
350
351
352 : )integers ( x( --- ) \ fill matrix with integers
353   ?x( dup )dimensions swap F! 0 \ a( C R
354   do dup 0 do
355     over j 1 _)) Fdup F! F! 0 F+
356     loop loop 2drop Fdrop reduce-error-code ;
357
358 *IF .{ 3 Test of matrix utility words } cr
359 value x( ' x( >body , 0 , here 0 , b/Float , 5 , 4 ,
360 here to x( here swap ! 4 5 = 2 = cells allot
361 x( 0 0 )) 7.7e
362 x( )list x( )integers x( )list forget x(
363 [then]
364
365 \ *** Creation of matrices *** \
366
367 : build-structure ( F R C pFa addr --- a( )
368   \ form a matrix at addr generating the new a(
369   descriptors cells+ >r \ pFa r: new a( address
370     dup 0 \ pFa olda_or_zero, data pointer contents
371     r! 2pick ! \ pFa olda_or_zero, place new x( pointer
372     r! backlink: ! \ pFa placed olda_or_zero
373     :r! pFa: ! \ place pFa:
374     r! dup check: ! \ a( address for checking purposes
375     !m: 0!r! ! \ also check:
376
```

Your editing view is not suitable for a presentation.

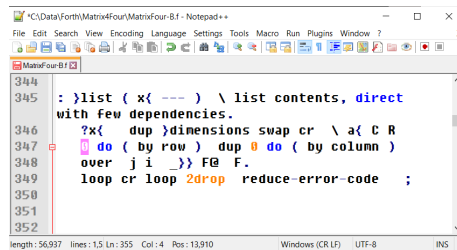




# Window Sizing

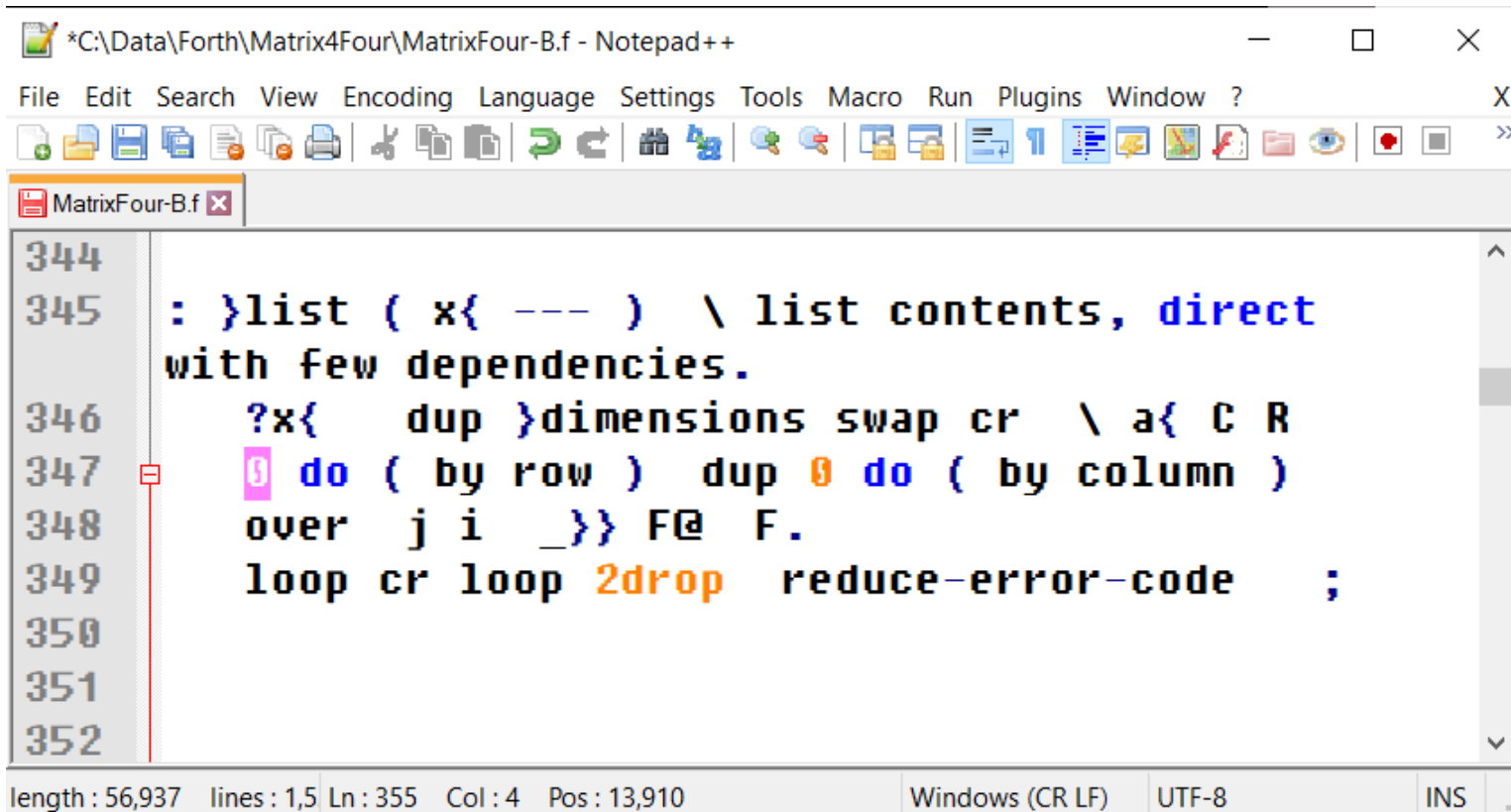
Set a small window on your computer.

Limit the view to one idea.

A screenshot of a Notepad++ window titled "C:\Data\Forth\MatrixFour\MatrixFour-81 - Notepad++". The window displays Forth code with line numbers 344 through 352. The code defines a word named 'list' that takes a variable 'x' and lists its contents. It uses 'direct' to show the stack and 'with few dependencies' to show the code. The code includes 'dup', 'dimensions', 'swap', 'cr', '\ a{ C R', 'do ( by row ) dup 0 do ( by column )', 'over j i \_>> F@ F.', 'loop cr loop 2drop reduce-error-code ;'. The status bar at the bottom shows "length: 56,937 lines: 1,5, Ln: 355 Col: 4 Pos: 13,910 Windows (CR LF) UTF-8 INS".

```
344  
345 : }list ( x{ --- ) \ list contents, direct  
with few dependencies.  
346 ?x{ dup }dimensions swap cr \ a{ C R  
347 { do ( by row ) dup 0 do ( by column )  
348 over j i _>> F@ F.  
349 loop cr loop 2drop reduce-error-code ;  
350  
351  
352
```

# Zoom Expands It To Full Screen



```
*C:\Data\Forth\Matrix4Four\MatrixFour-B.f - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
MatrixFour-B.f
344
345 : }list ( x{ --- ) \ list contents, direct
with few dependencies.
346 ?x{ dup }dimensions swap cr \ a{ C R
347 do ( by row ) dup 0 do ( by column )
348 over j i _}} F@ F.
349 loop cr loop 2drop reduce-error-code ;
350
351
352
length : 56,937 lines : 1,5 Ln : 355 Col : 4 Pos : 13,910 Windows (CR LF) UTF-8 INS
```

Now your idea is clear.

# Get The Most From Zoom

Simplify your background.

Get the best lighting.

Use a webcam showing your project.

# Speaking Of Your Face

Video lighting is a multi-faceted topic;  
here, just the very basics.

# Speaking Of Your Face

Video lighting is a multi-faceted topic; here, just the very basics.

Be sure your face is clearly illuminated by more than room lights.

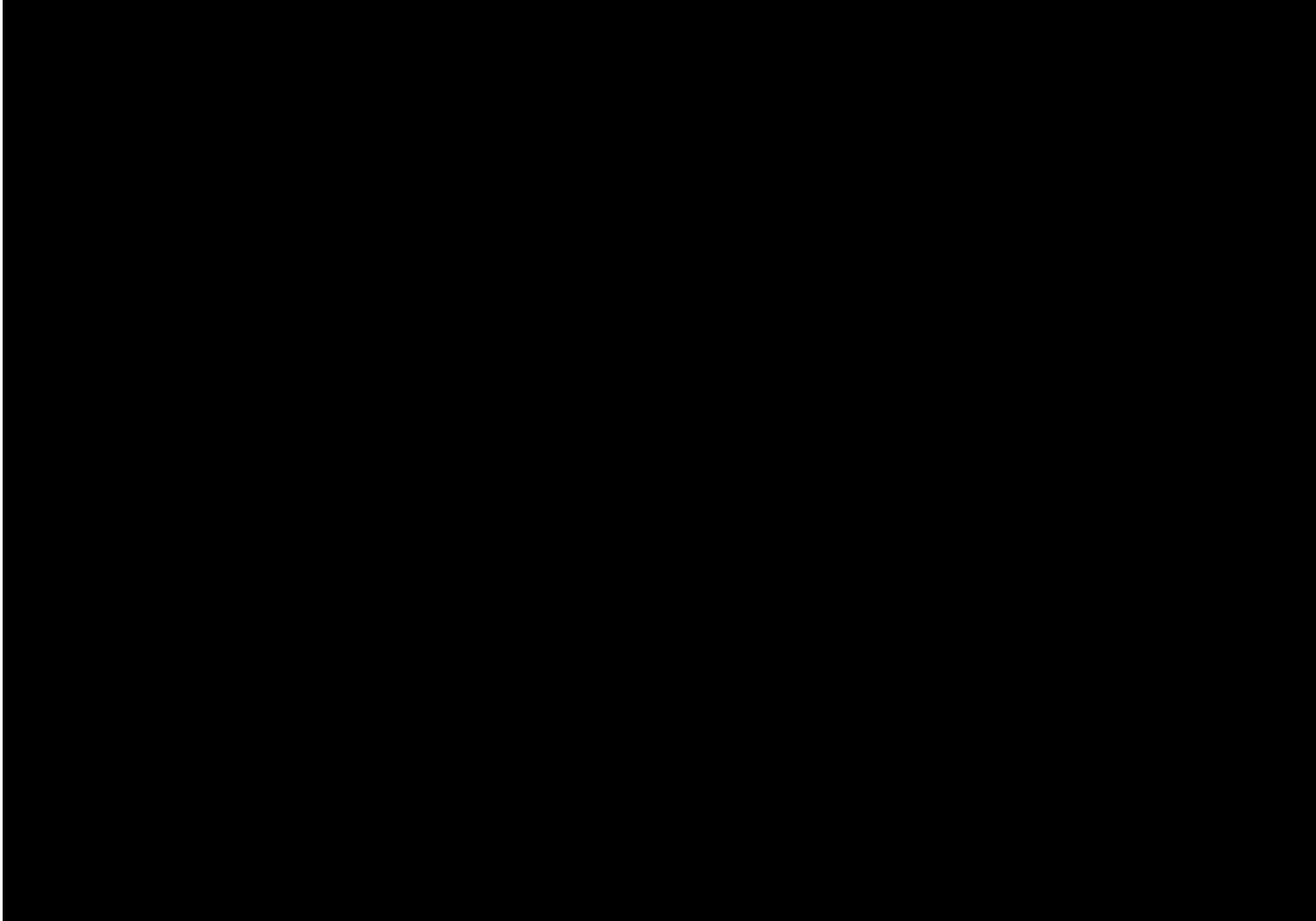
# Speaking Of Your Face

Video lighting is a multi-faceted topic; here, just the very basics.

Be sure your face is clearly illuminated by more than room lights.

And simplify or eliminate the background.

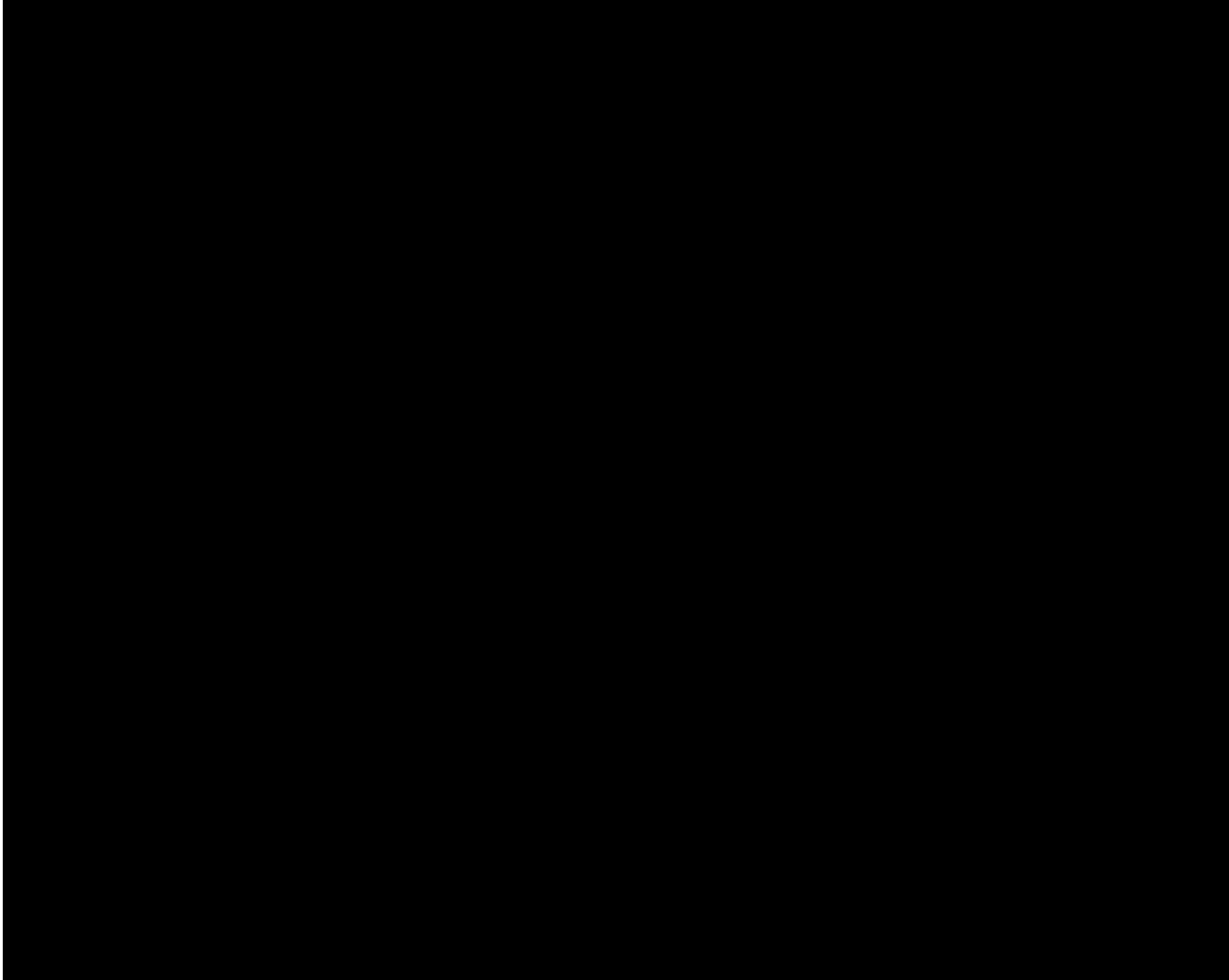
# Get The Most From Zoom Video







# Show Your Projects



# Live Streaming Software

Multi-stream, slides, desktop, video:

- X-Split Broadcaster; \$5/month.
- OBS, Open Broadcaster Studio, Free

Support "You have 1 new message" OPEN MESSAGE

(2/2) X



Scene Welcome

1



Transition Fade: 700ms



Integrated Webcam



Welcome

Slides

Demonstration

Scene 4

Scene 5

+

Add Source Copy Paste Remove Rename Settings



Support "You have 1 new message" OPEN MESSAGE (2/2)

PowerPoint Slide Show - [Forth2020-Presentation-03pptx.pptx] - PowerPoint

How To Build Forth Presentations

Forth2020  
Nov. 13, 2021  
Bill Ragsdale

Slide 1 of 38



Scene Slides

1 [dropdown] [volume icon] [mute icon]

Transition Fade: 700ms [dropdown] [star icon] [dropdown]

- Window "PowerPoint Slide Show - [SVFIG 2D Data Structure-02.pptx] - PowerPoint" in "po..." [eye icon]
- Integrated Webcam [lock icon] [eye icon]

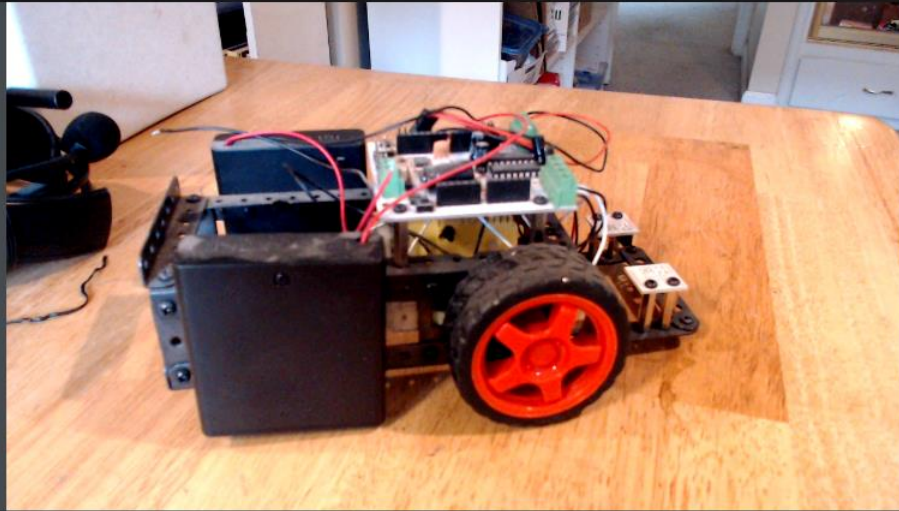
Add Source Copy Paste Remove Rename Settings [folder icon] [lock icon] [refresh icon] [list icon] [down icon] [up icon]

Welcome Slides Demonstration

Scene 4 Scene 5 +

Support "You have 1 new message" OPEN MESSAGE

(2/2) X



Scene Demonstration

1



Transition Fade: 700ms



HD Pro Webcam C920



- Welcome
- Slides
- Demonstration
- Scene 4
- Scene 5
- +

# Summary

Plan a structured presentation.

Set the proper depth of detail.

Keep code large for visibility

Optimize lighting and background.

Consider a hand camera for demos.

# Benefits

Maximize the information transfer.

Hold the viewers interest.

We are generating a long-term archive of Forth technology.

Get the best use of your resources now at hand.

# Conclusion

Today I used the SAFW model.

I STATED Zoom is our communication medium.

I ACCENTUATED communication factors.

I FOR INSTANCED offering many tools.

I WOUND UP with the benefits.



# References, later

For these slides and video go to:

- <http://www.forth.org/svfig/>

then “Past meeting slides, video, and notes.”

- - or -

- <http://www.forth.org/svfig/kk/kk.htm>