

OSMIC WORKSHOP # 7, 97.05.29

- ▽
- Kevo
- Kenichi
- Shu
- Yoyo

Kenichi's new diagrams 1

PARTS OF THE INTERFACE 4

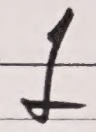
Where calculate new pointers? 6

WHAT GOES IN THE BACK END? 7

Miller's Law 8

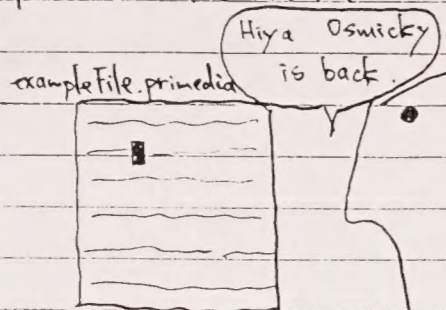
REASONS FOR MY SUGGESTED DIVISION 10

LAZY EDITOR & POINTERS 11

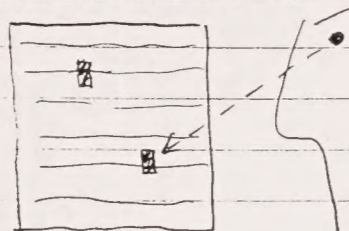


USER

1. Specifies the start - cursor position

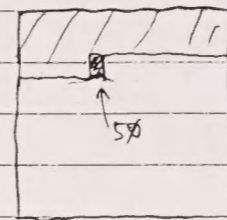


2. Specifies the end - cursor position



BACK - END

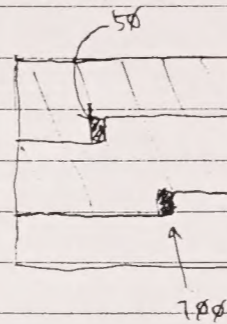
1.



Temporarily, this area is saved (or read as FileHandle) and checked for size (length) of the file (ex. size = 50 byte)

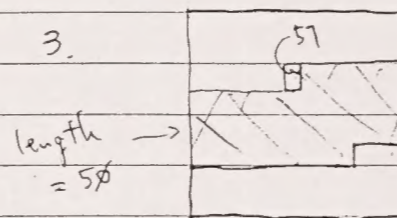
IS WHITESPACE COUNTED? IF NOT, WHITESPACE SHOULD BE FILLED WITH SOME CHARACTERS.

2.



Check for the cursor position in the same way and here comes the file size! (ex. size = 700 byte)

3.



Back-end specifies a pointer (start + length)

(ex. 57 + 50)

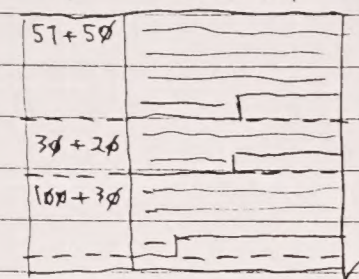
-> Extract the document with pointer, as it were.

4. Send the pointer to the front-end

(This means it's stored as a parameter of code)

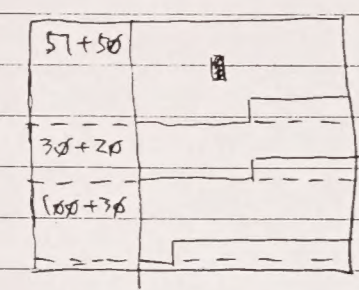
57 + 50

1. vv document



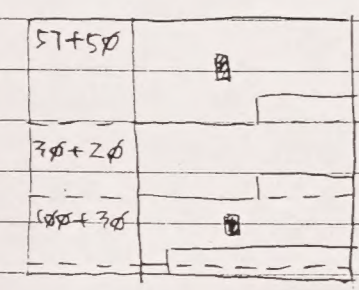
I'm about to delete some part.

2.



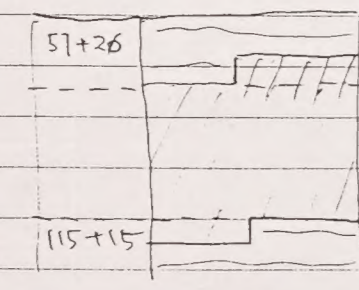
specified the start position

3.

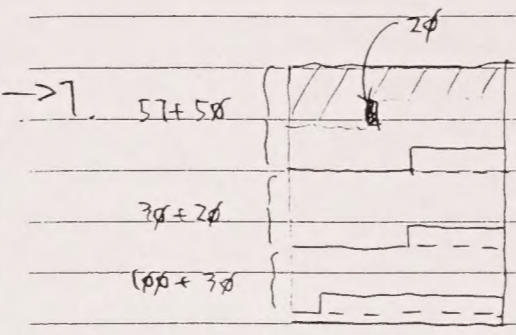


Then, the end position

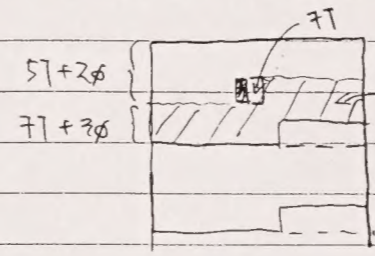
4.



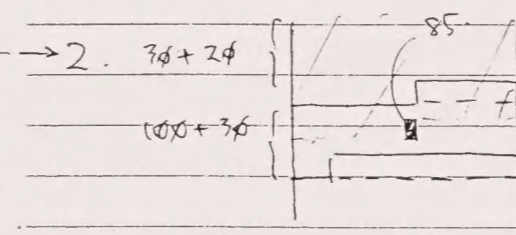
Disappeared



Back-end (Front-end?) temporarily saves these area as a file and checks for the file size (ex. size = 20 byte)

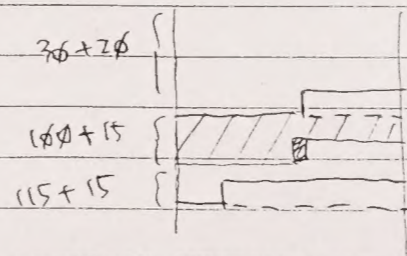


Then split the pointer (ex. $57 + 50$
 \downarrow
 $57 + 20$
 \downarrow
 $77 + 30$)



check in the same way (ex. size = 85 byte)

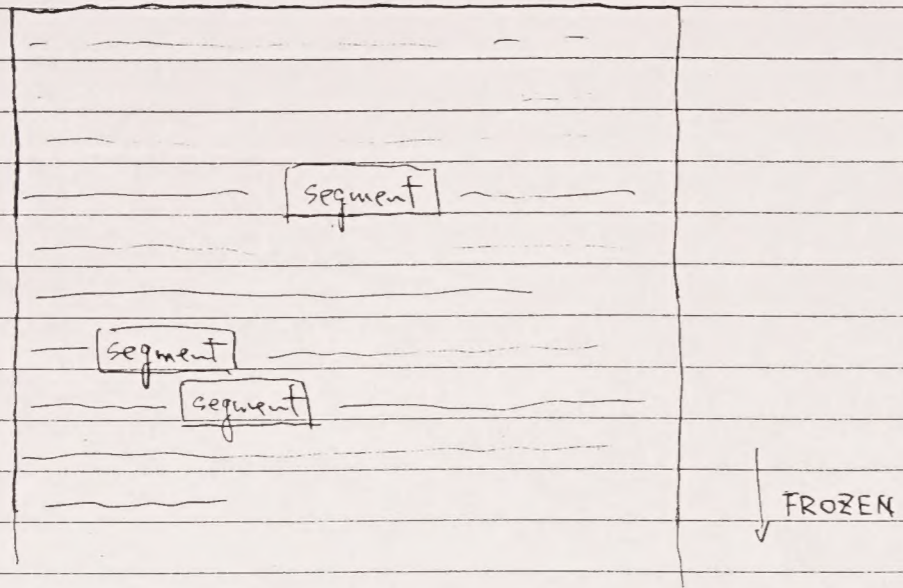
OOPS! THERE IS LARGE WHITESPACE? WE NEED TO FILL THESE AREA WITH SOME CHARACTERS.



Split the pointer (ex. $100 + 30$
 \downarrow
 $100 + 15$
 \downarrow
 $115 + 15$)

3. Discard the pointers between [start] - [] - [End]

exampleFile.primedia

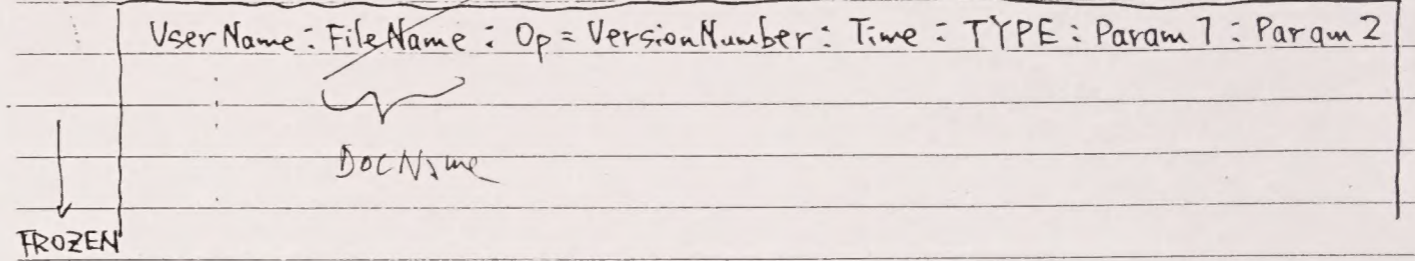


How do we specify the address dynamically?

Solution 1. Think the cursor position as the last char of file (= exampleFile.primedia)

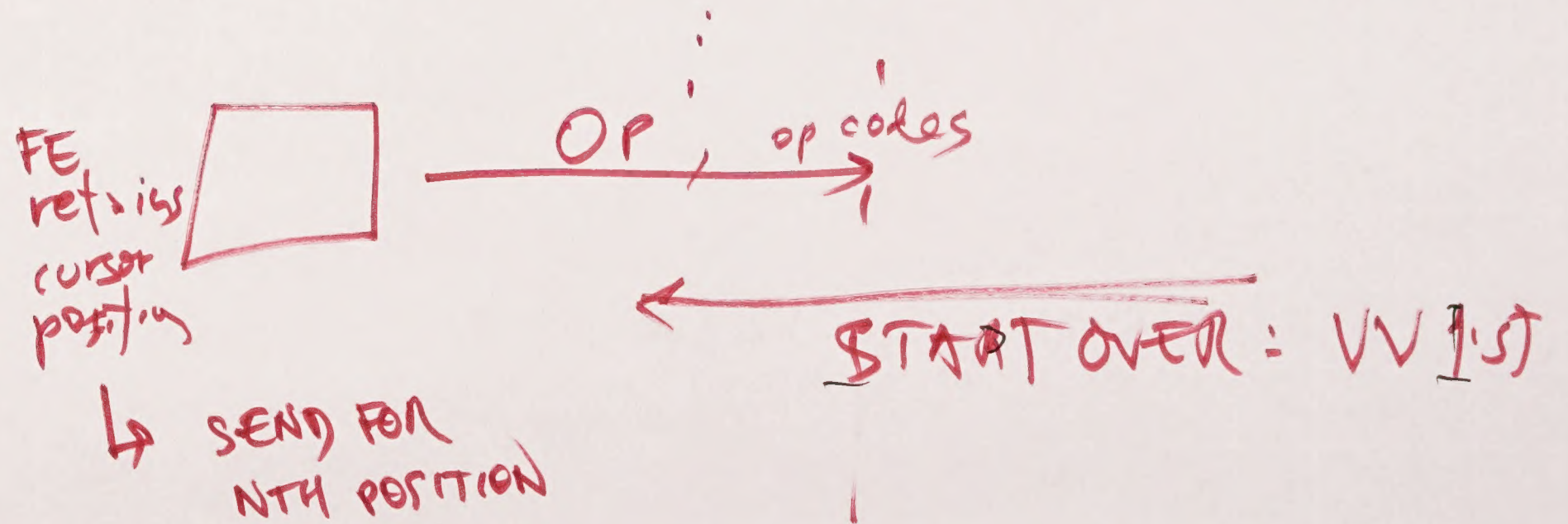
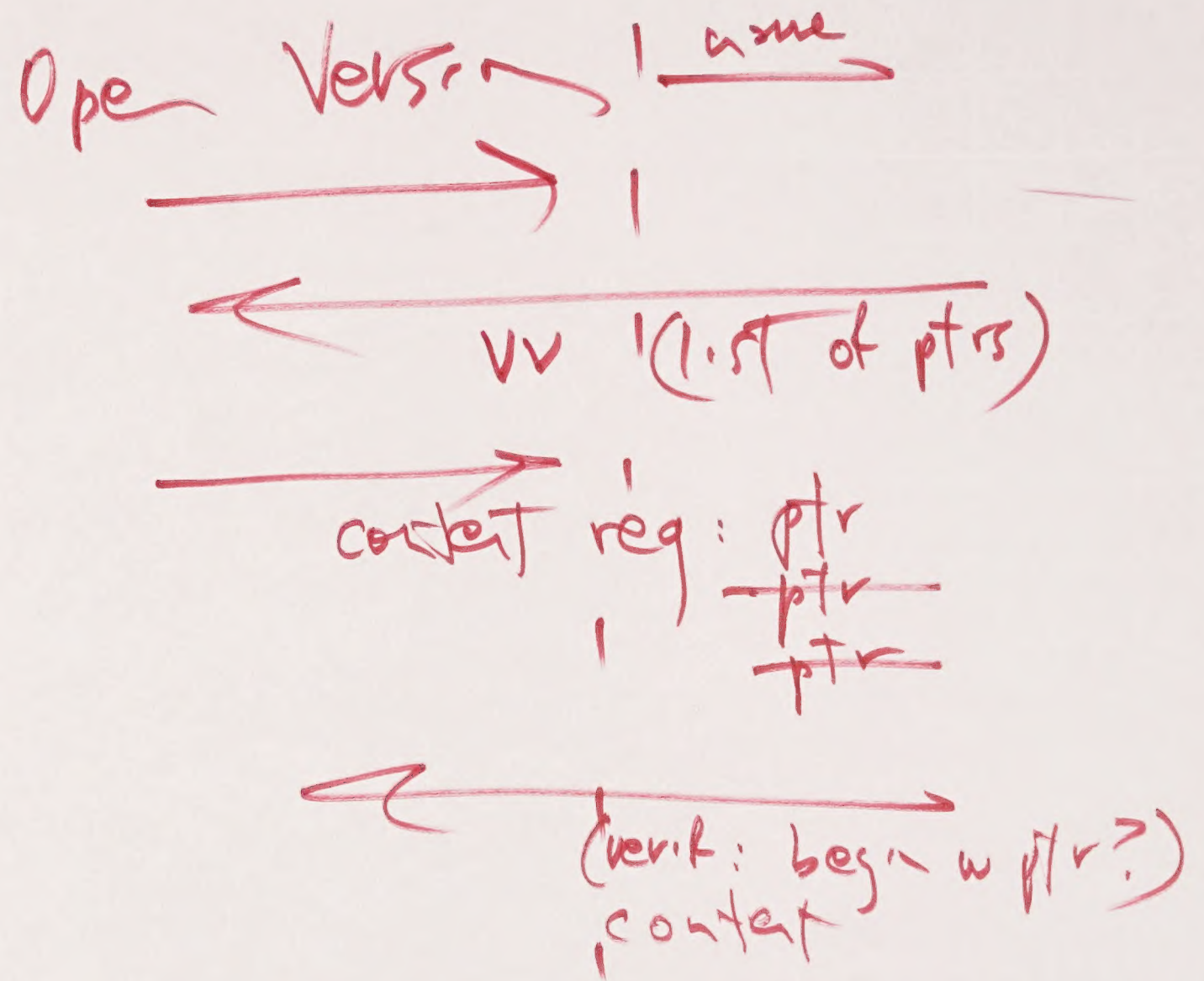
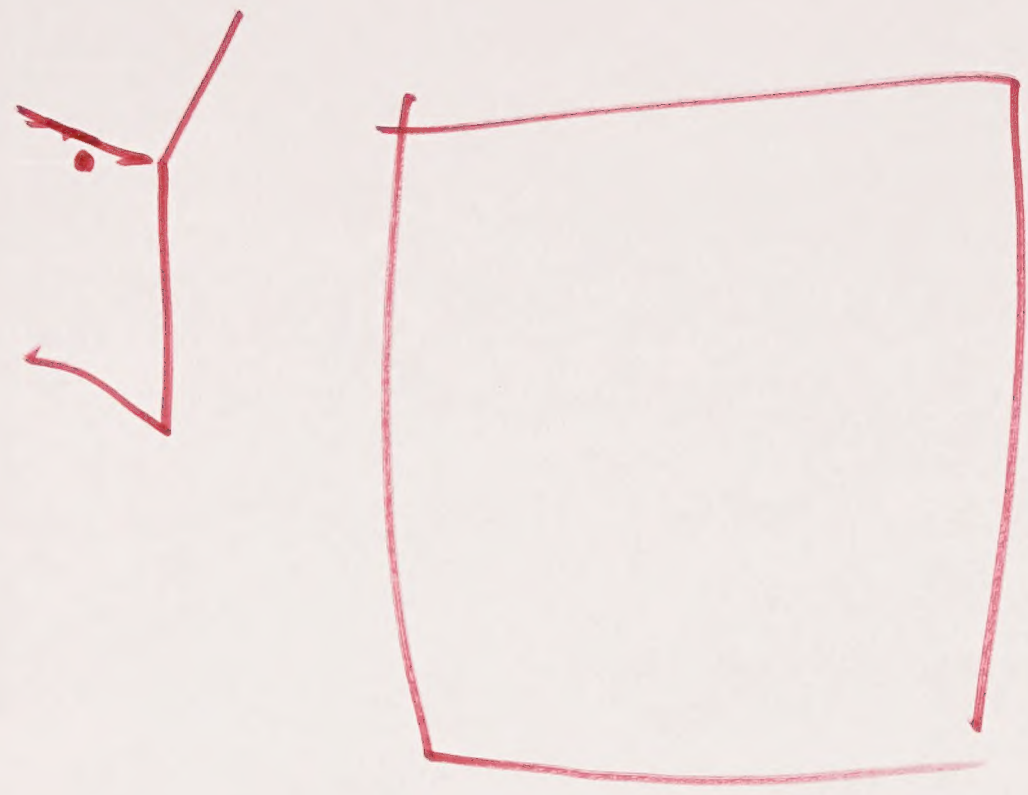
SEEMS NOT SO EFFICIENT WAY ?

exampleFile.code



THE INTERFACE = SOME MAIN PARTS

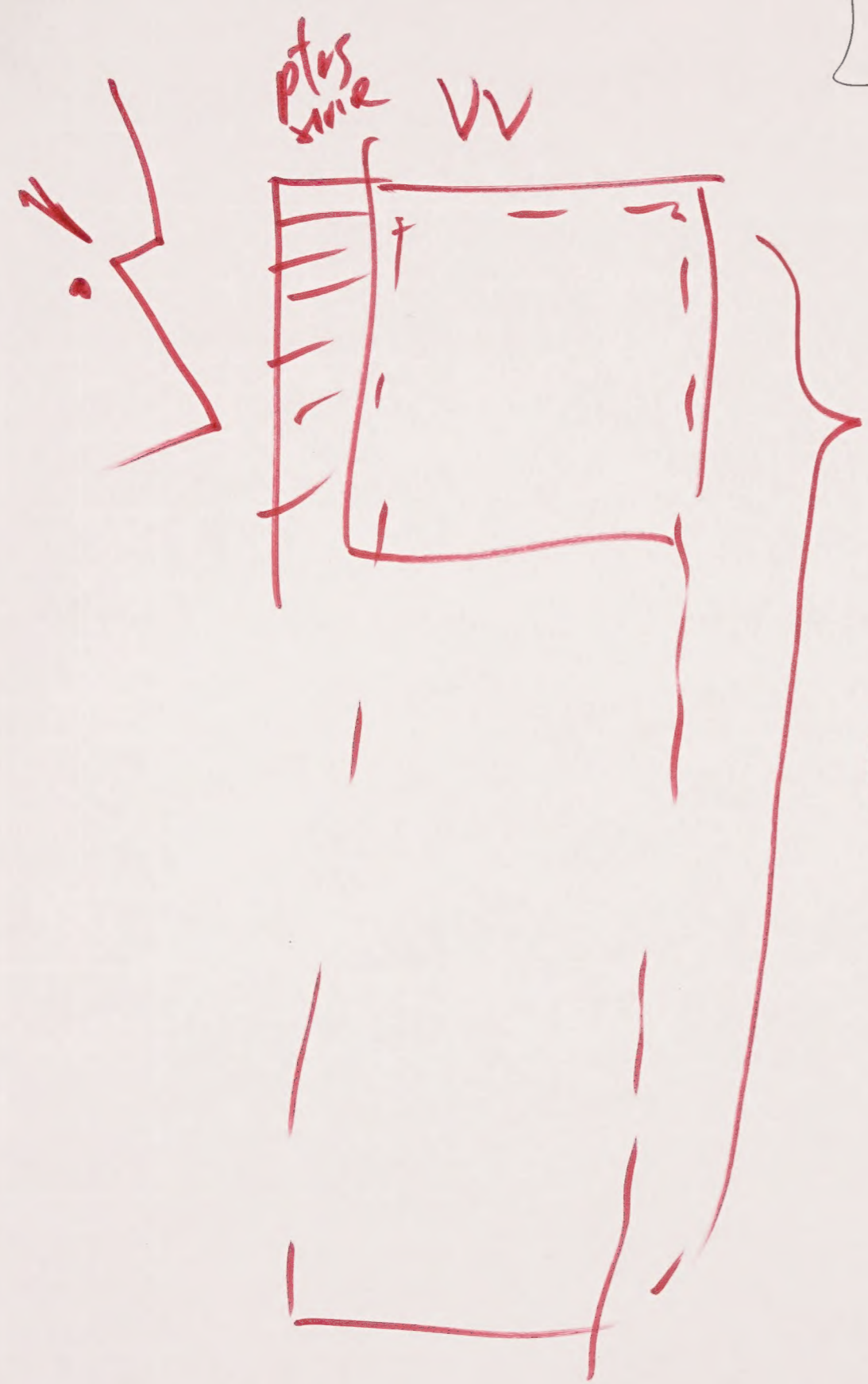
MAY 29 1997 4



Start of Session

MAY 29 1997

P/S



- ① pointer arrive
- ② front end asks for CONTENTS OF FIRST PTRS
- ③ scanning: back end sends more code

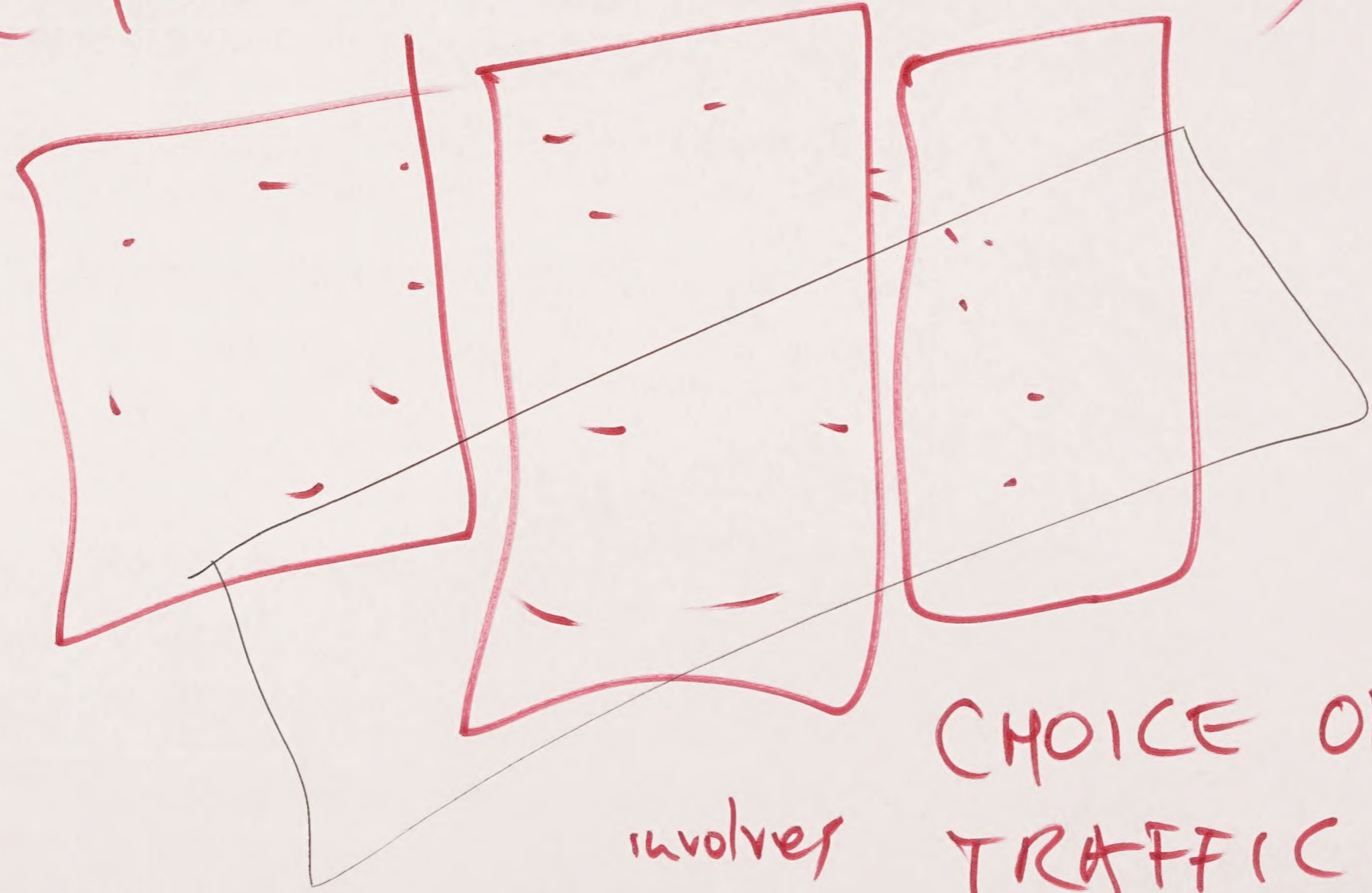
Where calculate new pointers?

There needs to be a module
that calculates these pointers for us

IT COULD BE IN BACK OR FRONT
MAYBE SHD BE IN BOTH EVENTUALLY

Where do we put it now?

WHAT GOES IN THE BACK END? (MODULARITY QUESTION)

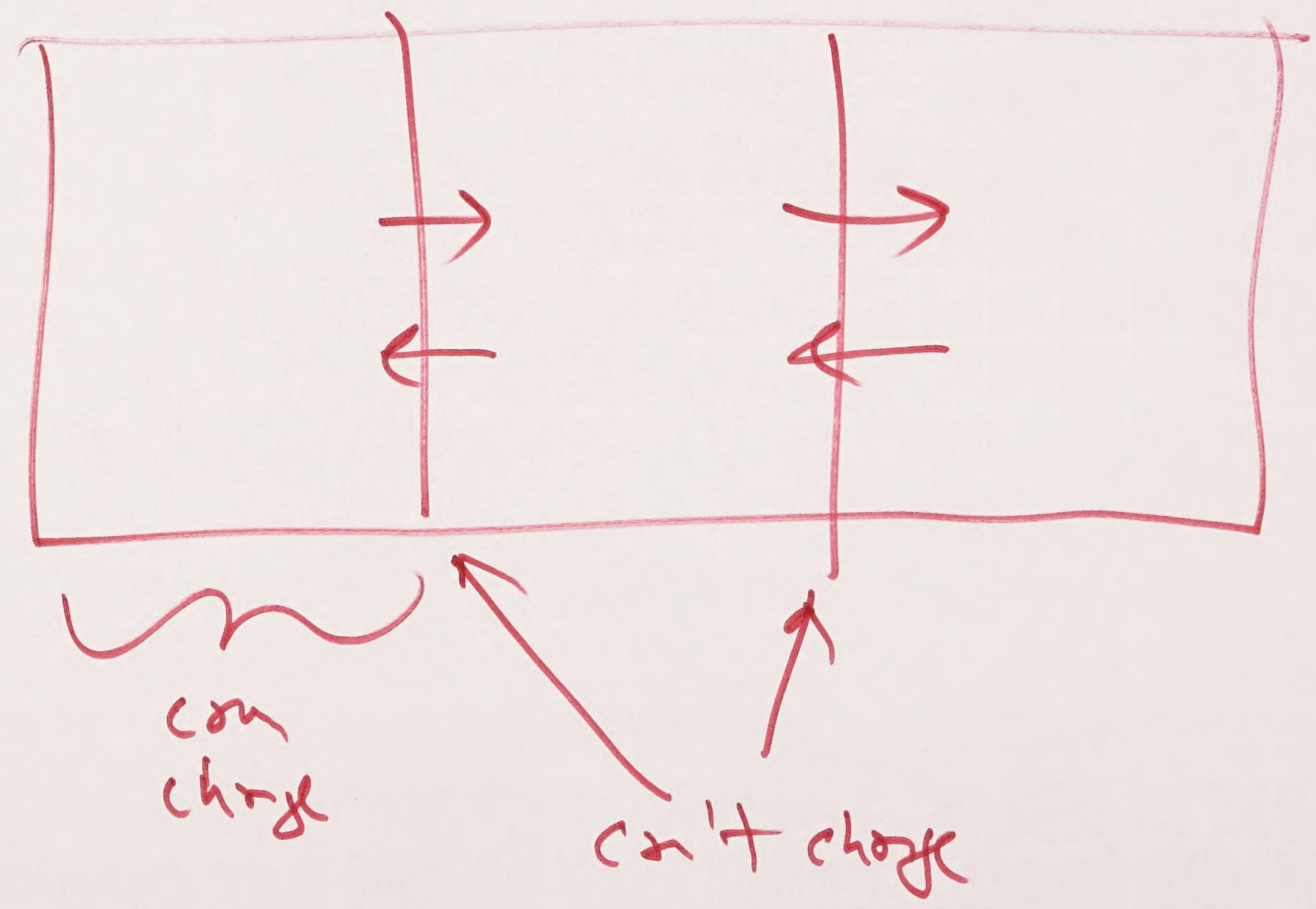


involves

- CHOICE OF BOUNDARIES
- TRAFFIC
- REALITY
- NEAT CODE

MILLER'S LAW =

"When you decide the interfaces,
you decide the tech forever."



If you put screen intersection in back end,

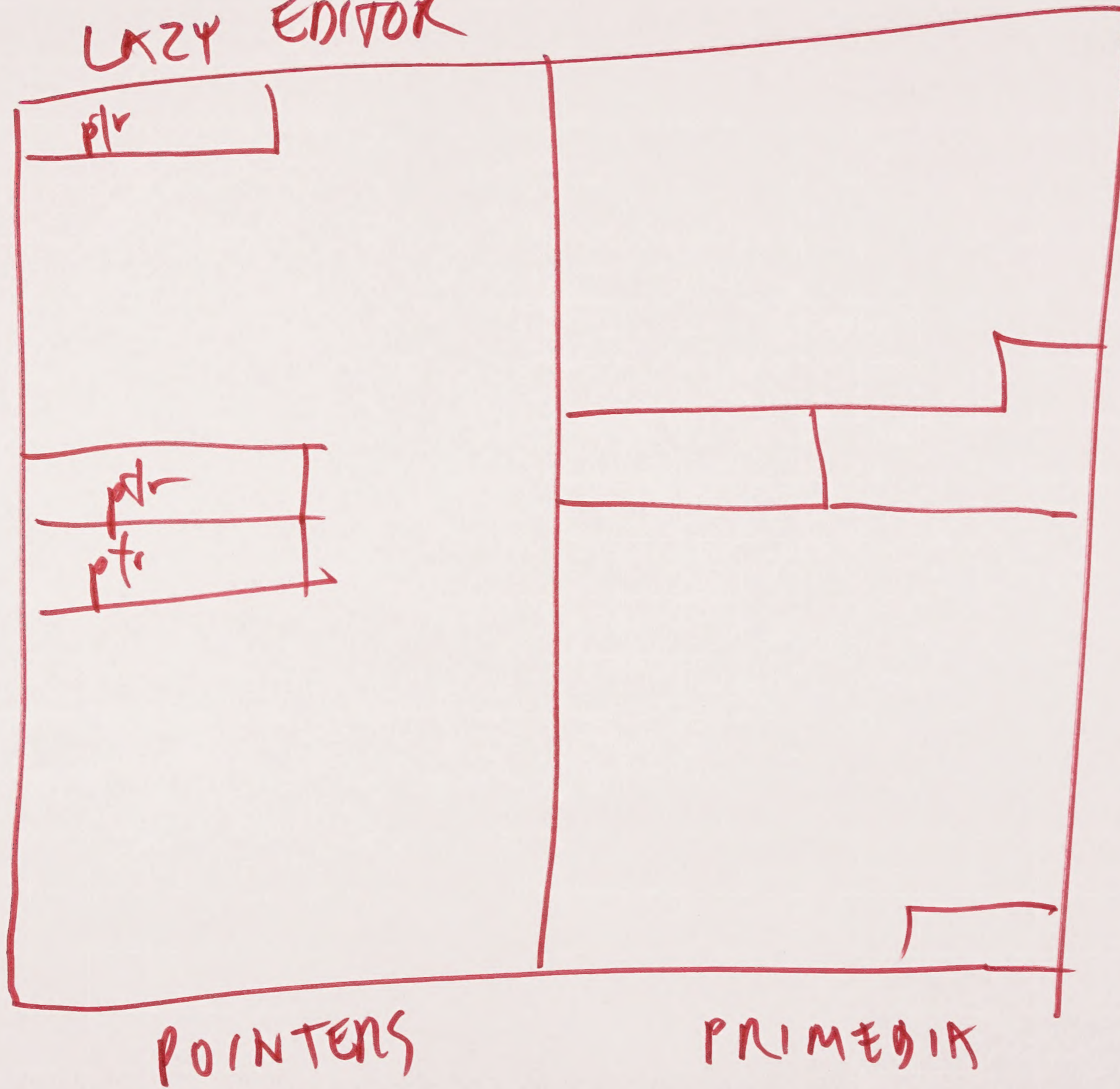
IT HAS TO KNOW ALL ABOUT:

screen mapping \leftrightarrow uv \leftrightarrow pointers



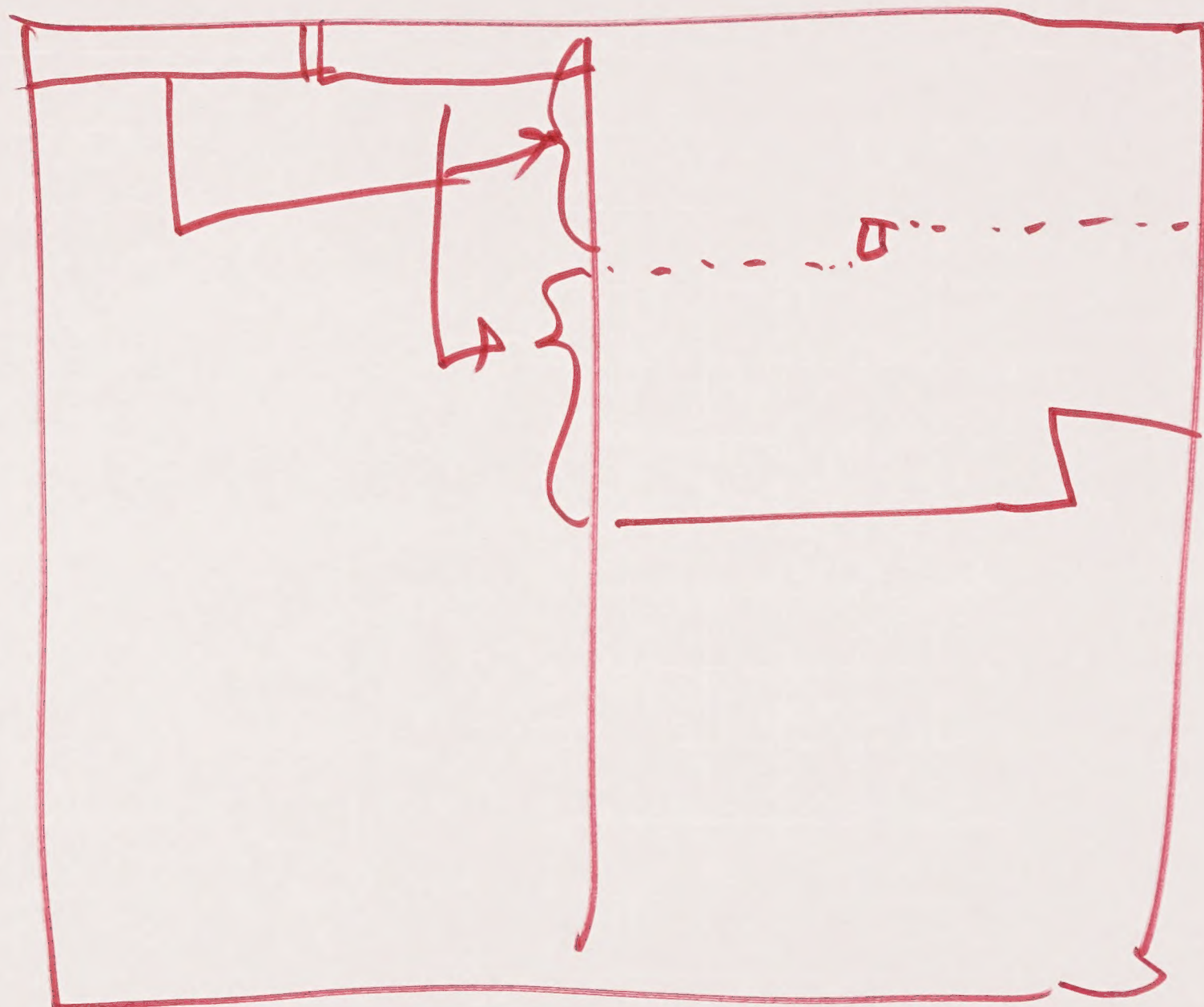
IF WE PUT IN FRONT END,
back end is
MUCH SIMPLER.

LAZY EDITOR



MAY 29 1997

7/12

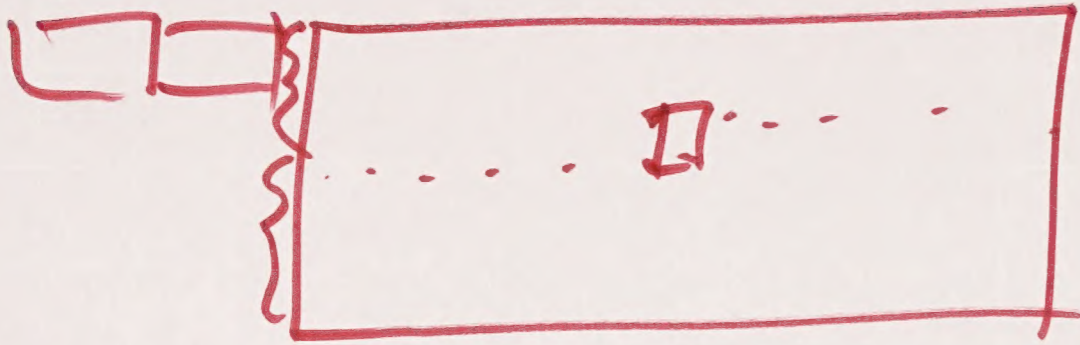


ANY CURSOR POSITION
SPECIFIES ~~POINTERS~~
POINTERS



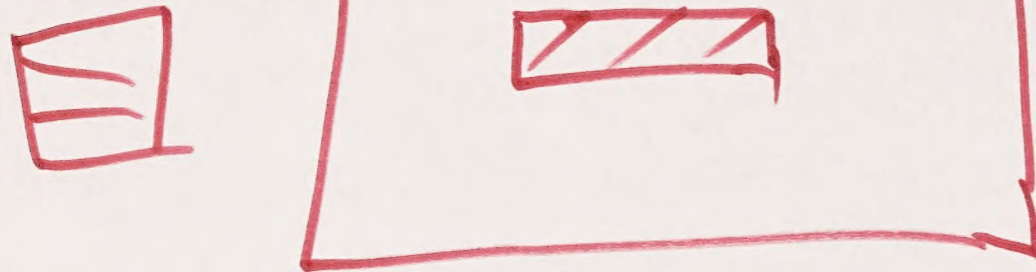
These pointers are
immediately ready
for an operation

Case 1



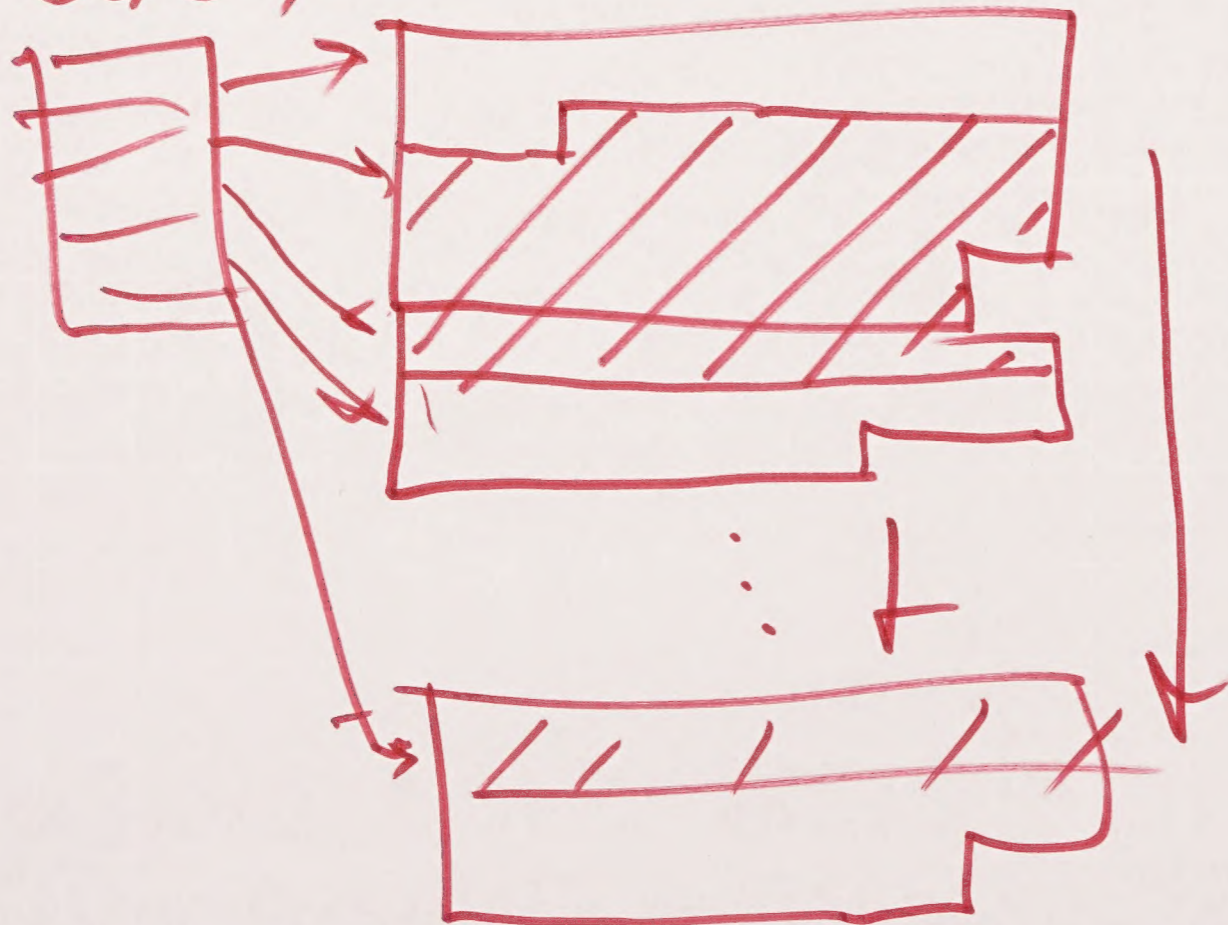
1 cut, 2 ptrs

Case 2



STRIPING = 2 cuts, 3 ptrs

Case 3



STRIPING ACROSS PIECE BOUNDARIES

4 ptrs

could go on to n ptrs



LIST OF PTRS GOES TO EDIT OP

IF DELETE

A TRANSLUDE

