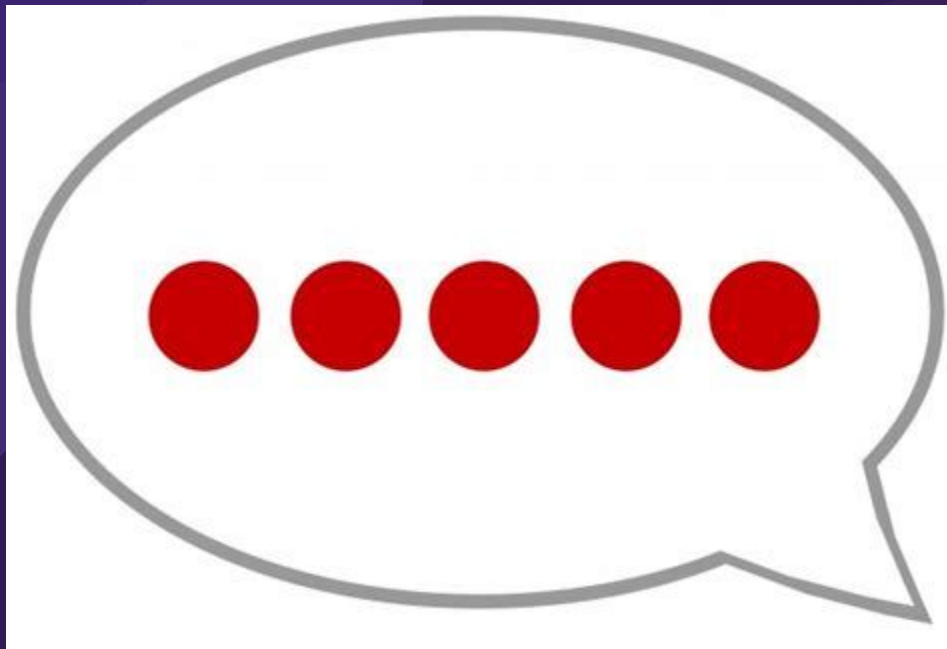


# Searching Your Elixir Code

---

Onorio Catenacci  
Elixir Developer  
Cover My Meds

# What We'll Discuss



# Text To Be Searched (T)

True Positive (P)

True Negative (N)  
False Positive  
False Negative

# Text To Be Searched (T)

True Positive (P)

Desired  
Result

Actual Result

True Negative (N)

False Positive  
False Negative

## How To Improve Our Chances?

### Reduce T In Which We're Searching

- Don't search some files
- Don't search some directories
- Don't search certain file types

## How To Improve Our Chances?

### Describe Search Term Better

- Wild card search term
- Describe context
- Is search term case-sensitive?
- Exclude non-matching

## How To Improve Our Chances?

### Use Specialized Tools Where Appropriate

- Programmer's Grep
- Pattern Matching Language

## You May Now Be Thinking

- Why can't I search in my editor?
- What about intellisense/LSP?



## Programmers' Grep

- What is a “grep”?
- How does a programmer's grep differ?

## Programmers' Grep

```
# By default skip the dependencies and test directories
ignore_dirs=("deps" "test" "config" "priv")
ign_dirs=()
for i in "${ignore_dirs[@]}; do
    ign_dirs+=("--ignore-dir=$i");
done
```

Exack.sh

## Programmers' Grep

```
# By default skip certain scripts
ignore_files=("is:mix.exs" "is:credo.exs" "is:.formatter.exs")
ign_files=()
for i in "${ignore_files[@]"; do
  ign_files+=("--ignore-file=$i");
done
```

Exack.sh

## Programmers' Grep

```
pager=less
```

```
file_types=elixir:ext:ex,exs,eex,heex
```

```
search_string=$1
```

```
ack "$search_string" --pager="$pager"
```

```
--type-set="$file_types" "${ign_dirs[@]}" "${ign_files[@]}"
```

```
--noenv "${@:2}"
```

Exack.sh



# Pattern Languages

## Regular Expressions:

“Now you have two problems” (with apologies to J. Zawinski)

Regular expressions are like javascript. Everyone wants RE but no one wants to write RE.

The regex we used is:

```
\b\d{1,8}\b[^\S]{10,100}?b(AKIALIARIAZICACOICTDCIDEIFLIGAIHIIAIIIDIIILINIKSIKYILAIMAIMD  
IMEIMIIIMNIMOIMSIMTINCINDINEINHINJINMINVINYOHIOKIORIPAIRIISCISDITNITXIUTIVAIVTI  
WAIWIIWVWY)\b\s\d{5}\b\b((\+|\b)[11][\-\.\ ])?\?(?)\b[\dOIZSB]{3,5}([\-\.\ ]\|) ?)[\dOIZSB]{3}[\-\.  
][\dOIZSB]{4}\b|[\w\.\=+@][\w\.\-]+\.[\w]{2,3}\b(birth|birthdate|birthday|dob|born)\W+(?:\w+\W+)  
{0,5}?(?:\d{4}\d{1,2})[\V-]\d{1,2}[\V-]\d{4}\d{1,2}))\b\b([0-8]\d{2}|7([0-6]\d))([-\]?[s]{1})\d\d\  
d{4}\b|(?:(?:5[1-5][0-9]{2}|222[1-9]|22[3-9][0-9]|2[3-6][0-9]{2}|27[01][0-9]|2720)[0-9]{12})\b([4\  
d{3}][s]\d{4}[s]\d{4}[s]\d{4}[4]\d{3}[-]\d{4}[-]\d{4}[-]\d{4}|[4]\d{3}[.]\d{4}[.]\d{4}[.]\d
```

19

```
{4}|[4]\d{3}\d{4}\d{4}\d{4}\d{4})\b|3[47][0-9]{13}\d{3}-\d{2}-\d{4}|(?:\d{1,5}( 1V[234]))?(\x20[A-Z  
][a-z])+ )|(P\.\O\.\ Box \d{1,5}))\s{1,2}(?:((APT|B LDG|DEPT|FL|HNGR|LOTI|PIER|IRM|S(LIP|  
PC|T(E|OP))|TRLR|UNIT)\x20\w{1,5})|(BSMT|FRNT|LBBY|LOWR|OFC|PHIREAR|SIDE|UPPR)\.?)  
[s]{1,2}))?(?:[A-Z]([a-z])+(\.?)\x20[A-Z]([a-z])+){0,2})\., \x20(?:A[LKSZRAP]|C[AOT]|D[EC]|F[  
LM]|G[AU]|HII|ADL N]|K[SY]|LA|M[ADEHINOPST]|N[CDEHJMVY]|O[HKR]|P[ARW]|R|S[CD]  
|T[NX]|U|V[AIT]|W[AIVY])\x20(?:\d{5}(-\d{4})?)|(?:\d{1,5}( 1V[234]))?(\x20[A-Z]([a-z])+ )|(P\.  
O\.\ Box \d{1,5}))|A-Z0-9<]{9}[0-9]{1}[A-Z]{3}[0-9]{7}[A-Z]{1}[0-9]{7}[A-Z0  
-9<]{14}[0-9]{2}|A-Z9]{5}[0-9]([05][1-9]|16)[0-2])(0[1-9]|12)[0-9]3[01])[0-9][A-Z9][0-9][  
A-Z0-9]([0-9]{2})?
```

# Pattern Languages

A few caveats:

- You need to do extra work to match over line boundaries
- Your description must be very exact—spaces and all!
- Upper/lower case matter for matching
- A badly formed RE can literally run in  $O(n!)$  time
- RE's aren't amenable to maintenance
- RE do not support recursion (no HTML)



Pattern Languages

# A Better Alternative

# Pattern Languages

## A Taste Of Rosie Pattern Language

```
module_name = {~ [:upper:]{1}{[:alnum:]}"/."}+ ~}
```

```
-- test module_name accepts " Test "
```

```
-- test module_name accepts "Test"
```

```
-- test module_name accepts " Test.Submodule "
```

```
-- test module_name accepts "Test.Submodule"
```

Elixir.rpl

# Pattern Languages

## A Taste Of Rosie Pattern Language

*grammar*

```
module_start = ^[:space:]* "defmodule" module_name "do"  
[:space:]*$
```

*in*

```
module = ~ !comment_char module_start
```

*end*

Elixir.rpl



# Resources

Programmer's Greps:

Ack: <https://beyondgrep.com>

Ag (The Silver Searcher): [https://github.com/ggreer/the\\_silver\\_searcher](https://github.com/ggreer/the_silver_searcher)

RipGrep: <https://github.com/BurntSushi/ripgrep>

Pattern Language:

Rosie Pattern Language: <https://rosie-lang.org>

RegEx

Onorio Catenacci

Twitter: @OldDutchCap

Email: [catenacci@ieee.org](mailto:catenacci@ieee.org)

Code Used In Talk:

[https://github.com/OnorioCatenacci/ex\\_search](https://github.com/OnorioCatenacci/ex_search)