Audit

Ian McCormack

2023-01-24

Contents

Chapter 1

Introduction

Chapter 2

ABI Distribution

The maximum number of ABIs defined in a single file.

```
by_count <- late_abis %>% group_by(crate_name) %>% summarize(n=n())
print(max(by_count$n))
```

```
## [1] 4
```

The number of ABI occurrences in files.

Chapter 3

Improper Type Errors

We observed 16,771 type errors in foreign bindings. There were 480 unique errors which occurred for 229 unique crates. Of these crates,

There were 46 crates that globally disabled the `improper_ctypes` lint, there were 12 crates that globally disabled the `improper_types_definitions` lint, and there were 10 crates that globally disabled both lints.

There were 51 crates that selectively disabled the `improper_ctypes` lint, there were 9 crates that selectively disabled the `improper_types_definitions` lint, and there were 0 crates that selectively disabled both lints.

When combined, there were 97 crates that disabled the `improper_ctypes` in some fashion, there were 21 crates that selectively disabled the `improper_types_definitions` lint in some fashion, and there were 10 crates that disabled both lints in some fashion.

This provides us with a sample of 107 crates that hid improper type errors.

```
## # A tibble: 4 x 7
##   Category          `# Crates` `# Unique Errors` Total        Mean  St.Dev Max
##   <chr>             <chr>      <chr>             <chr>        <chr> <chr>  <chr>
## 1 Foreign Functions 144 / 93   14 / 14           16140 / 144~ 112.~ 473.6~ 5375~
## 2 Rust Functions    78 / 15    13 / 9            437 / 195    5.6 ~ 15.8 ~ 136 ~
## 3 Static Items      15 / 6     8 / 4             194 / 171    12.9~ 25.3 ~ 99 /~
## 4 All               229 / 107  35 / 27           16771 / 148~ 73.2~ 385.1~ 5474~
```

```
## # A tibble: 16 x 7
##    Type    Error  Unique `Foreign Funct~` `Foreign Items` `Rust Functions` Total
##    <chr>   <chr>   <int>            <dbl>           <dbl>            <dbl> <dbl>
##  1 Adt     EnumN~      2                4               0                0     4
##  2 Adt     EnumN~      3               29               7               21    57
##  3 Adt     Struc~      3             8286             161                7  8454
```

```
##  4 Adt     Struc~      1              2              0              0     2
##  5 Adt     Union~      3            912              2             10   924
##  6 Adt     OnlyP~      1              1              0              0     1
##  7 Array   --          2             21              0              0    21
##  8 Char    --          2             19              0              0    19
##  9 Dynamic Dyn         3              3              0              4     7
## 10 FnPtr   --          3            295              1              0   296
## 11 Int     Num12~      2              1              0             66    67
## 12 Slice   --          2              1              0              1     2
## 13 Str     --          2              0              0              4     4
## 14 Tuple   --          3              7              0             24    31
## 15 Uint    Num12~      3           4856              0             58  4914
## 16 All     --         35          14437            171            195 14803
```

```
# large <- function(x){
#   paste0('\textbf{', x, '}')
# }
# addtorow <- list()
# addtorow$pos <- list(0, 0)
# addtorow$command <- c("& & & \\multicolumn{4}{c}{\\# Occurrences} \\\\\n",
# "Category & \\# Crates & \\# Unique & Total & Mean & St.Dev. & Max \\\\\n")
# xt_counts = xtable(
#   count_summary,
#   type = "latex",
#   digits=c(0,0,0,0,0,1,1,0),

# )

# align(xt_counts) <- "llcc|cccc"
# print(xt_counts,
#       include.rownames = FALSE,
#       add.to.row = addtorow,
#       file = "../latex/err_counts.tex",
#       include.colnames = FALSE,
#       sanitize.colnames.function = large,
#       floating=FALSE,
#       latex.environments=NULL
# )

# addtorow <- list()
# addtorow$pos <- list(0, 0)
# addtorow$command <- c("& & & \\multicolumn{4}{c}{\\# Hidden Occurrences} \\\\\n",
# "Error Category & Error & \\# Unique & Foreign Functions & Foreign Items & Rust Func
# xt_type_counts <- xtable(
#   filtered,
#   type = "latex",
```

```
#   digits=c(0,0,0,0,0,0,0,0),
#   floating=FALSE,
#   latex.environments=NULL
# )
# align(xt_type_counts) <- "lllc|cccc"
# print(xt_type_counts,
#       include.rownames = FALSE,
#       add.to.row = addtorow,
#       file = "../latex/err_type_counts.tex",
#       include.colnames = FALSE,
#       sanitize.colnames.function = large,
#       floating=FALSE,
#       latex.environments=NULL
#     )
```

Chapter 4

Quantitative Results

We began with a snapshot of the crates.io database taken on 12/20/2022, which contained 102,359 crates. After filtering out crates without any valid published versions, 98,501 crates remained. We ran our linting tool on this population. Prior to linting, we found that the latest version of 57 crates contained an empty archive when downloaded from crates.io. For crates with non-empty archives, 85,694 passed the early linting stage, while 12,807 failed it. Of those that passed the early linting stage, 83,533 passed the late linting stage, while 2,161 failed it. We use the subset of crates that passed both linting stages as the basis for our study, which comprises **82% of all crates** in the database. Though 474 crates that passed the late linting stage failed to compile, we still include these in our sample to avoid excluding crates that have valid Rust code but are missing dependencies that cargo cannot provide.

4.1 ABI Distribution

```
## # A tibble: 7 x 2
##   abi                `sum(count)`
##   <chr>                     <dbl>
## 1 C                         84310
## 2 C-unwind                      1
## 3 platform-intrinsic           46
## 4 Rust                      95021
## 5 stdcall                     128
## 6 system                      415
## 7 win64                         2
```

Both the early and late lints recorded the number of occurrences of each ABI for the rust functions, foreign functions, and static items declared in each crate. There were 4,155 crates that only appeared in results from the early lint. How-

ever, there were 3,437 crates that had one or more foreign ABIs that were *only* detected in the early lint.

For qualitative analysis, we focus on the subset of improper type errors that were explicitly hidden by developers that disabling the corresponding lint.