# Clean up your EDA flows with tclint

Noah Moroze

me@noahmoroze.com

Open Source EDA BoF @ DAC '24 • June 25, 2024

#### tclint checks Tcl scripts for issues

#### place.tcl

```
if {$optlevel > 1} {
  place_design -effort high
} elseif {optlevel > 0} {

place_design -effort low
}

check_place -out "place.rpt"

save_design "design.def"
```

```
$ tclint place.tcl

place.tcl:3:9: syntax error: invalid
bareword in expression: optlevel

place.tcl:4:1: expected indent of 2 spaces,
got 0 [indent]

place.tcl:7:1: invalid option '-out', did
you mean '-file' [command-args]
```

# Linting is SW best practice



Flake8



Verilator



**ESLint** 



Clippy



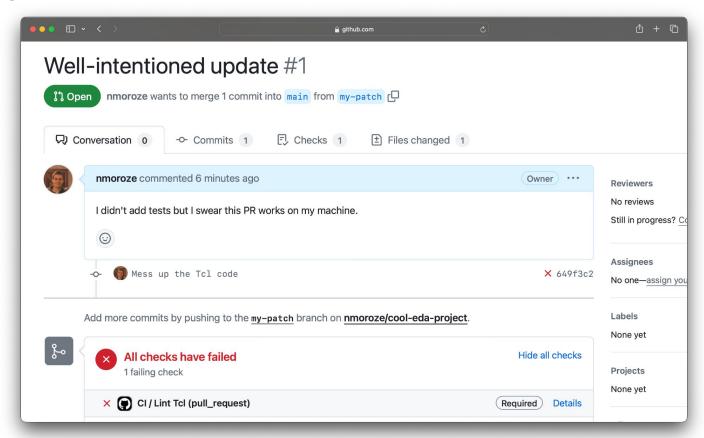
Vet

#### Linting is particularly valuable for EDA

#### place.tcl

```
1 if {$optlevel > 1} {
   place_design -effort high
3 } elseif {optlevel > 0} {
4 place_design -effort low
                                    Bugs hide in branches
5 }
7 check place -out "place.rpt"
                                    Runtime crashes can waste lots
8 save_design "design.def"
                                    of time
```

## Linting is particularly valuable for open source



## tclint is being used in Cl

- siliconcompiler/siliconcompiler
- The-OpenROAD-Project/OpenROAD
- foss-for-synopsys-dwc-arc-processors/toolchain





The OpenROAD Project

## tclint has caught bugs in the wild

```
@@ -66,10 +66,10 @@ proc iswithin { ADDRESS BASE LEN } {
66
     proc address info { ADDRESS } {
                                                                            66
                                                                                  proc address info { ADDRESS } {
67
                                                                            67
         foreach WHERE { FLASH RAM MMREGS XMEM UNKNOWN } {
                                                                                      foreach WHERE { FLASH RAM MMREGS XMEM UNKNOWN } {
                                                                            68
             if { info exists $WHERE } {
                                                                                         if { [info exists $WHERE] } {
70
                set lmt [set N [set WHERE]]
                                                                            70
                                                                                             set lmt [set N [set WHERE]]
                for { set region 0 } { $region < $lmt } { incr region } {</pre>
                                                                                             for { set region 0 } { $region < $lmt } { incr region } {</pre>
71
                                                                            71
72 -
                    if { iswithin $ADDRESS $WHERE($region, BASE)
                                                                            72 +
                                                                                                 if { [iswithin $ADDRESS $WHERE($region, BASE)
      $WHERE($region,LEN) } {
                                                                                  $WHERE($region,LEN)] } {
                        return "tWHFRF tregion".
                                                                            73
                                                                                                     return "$WHERE $region".
73
74
                                       ■■ siliconcompiler/tools/openroad/scripts/sc_write_images.tcl [☐
                                                                                                                            ☐ Viewed
75
                                @@ -141,7 +141,7 @@ proc sc_image_routing_congestion {} {
                        141
                                  sc image heatmap "Routing Congestion" \
                                                                                                 sc image heatmap "Routing Congestion" \
                                                                                      141
                                    "Routing" \
                        142
                                                                                      142
                                                                                                   "Routing" \
                                                                                                   "routing_congestion.png" \
                        143
                                    "routing congestion.png" \
                                                                                      143
                       144
                                    "routing congestion"
                                                                                      144 +
                                                                                                   "routing congestion" \
                        145
                                                                                      145
                                                                                      146
                        146
                        147
                                                                                      147
```

### Command plugins check tool-specific scripts

- Currently supports OpenROAD
- Dynamically generated by reading help strings

```
$ tclint-plugins make-spec openroad -o or-commands.json
$ tclint or_scripts/ --commands or-commands.json
```

## There are many features on the roadmap...

- Command plugins
- Auto-format
- Parser library

#### ... and you can shape its path!

#### **Download**

\$ pip install tclint

#### Run

- \$ cd my-eda-flow/
  \$ tclint .
- Give feedback!

Scan the QR code to see the Github repo Give it a star and/or open an issue! github.com/nmoroze/tclint

