

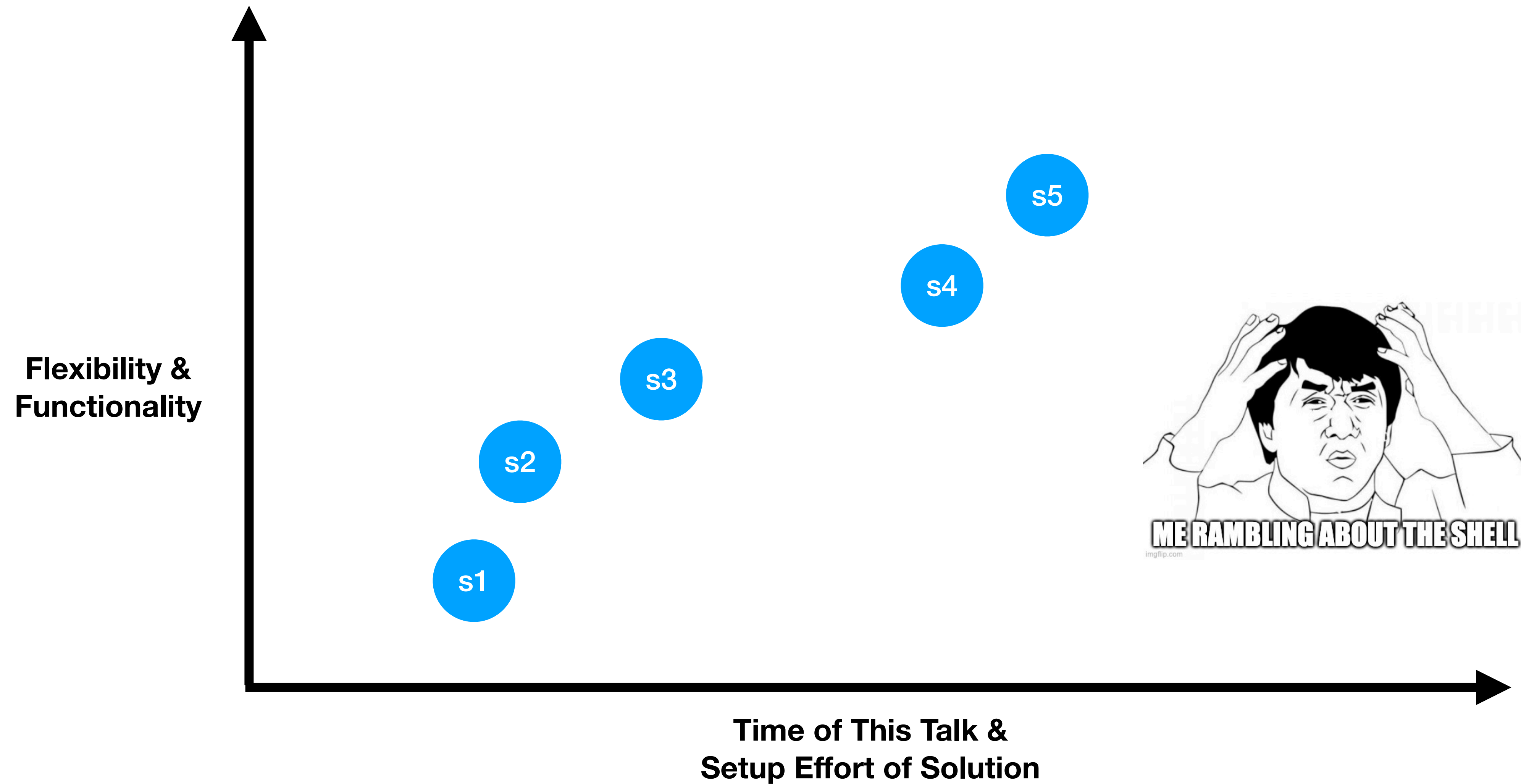
Organising Loads of **Kubeconfigs** Without Going Crazy

by Simon Bein ([@SimonTheLeg](#))



**“Hi, I’m Simon and I might have
too many kubeconfigs”**

Structure of This Talk



This is an opinionated talk

```
alias kc='kubectl'
```

Kubeconfig Basics

- Multiple clusters can be stored in the same kubeconfig using contexts
- Context can be switched using
`> kc config use-context abc`
- You cannot easily append new clusters, due to the structure of the file

```
# ~/.kube/config
clusters:
  - cluster: ...
  - cluster: ...
contexts:
  - context: ...
  - context: ...
users:
  - user: ...
  - user: ...
current-context: abc
```

s1 – Good Old Kubectl

- Multiple kubeconfigs can be switched via `$KUBECONFIG` or `--kubeconfig`
- You can merge multiple kubeconfig files into one

```
> KUBECONFIG=file1:file2 kc config view --flatten > merged
```

- Or split them

```
> kc config view --minify --flatten --context=context-1 > context-1
```



s2 – Wojciech's Genius 3 liner

Demo

s2 – Wojciech's Genius 3 liner

```
# .bashrc/.zshrc
kcsw() {
    local searchdir=${1:-./kubeconfigs}
    kubeconf=$(find $searchdir -type f | fzf)
    export KUBECONFIG=kubeconf
}
```

s3 – ahmetb/kubectx

★ 13.2k

Demo

s4 – sbstp/kubie

★ 1.2k

Demo

s5 – simontheleg/konf-go

★ 15

Demo

So how is it so fast?

**Because we don't need to
spawn a new shell**

Then.. Why don't we need to spawn a new shell?

- A child process cannot make changes to its parent process
- So you cannot set **\$KUBECONFIG** in shell with the binary itself...right?
- There is an exception: bash/zsh functions



What exactly is konf?

konf-go

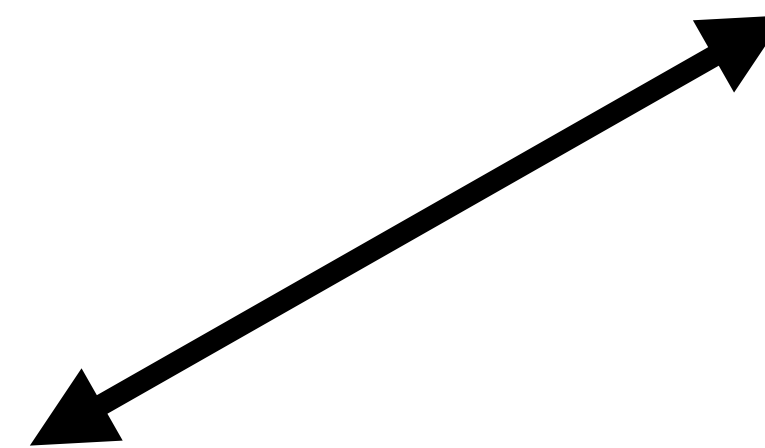
Go binary

Does the heavy lifting in the
background

konf

Bash/zsh function

calls konf-go, takes its output
and sets \$KUBECONFIG



```
> konf set simons-fun-cluster
```


Demo Time

konf is now in beta @ [SimonTheLeg/konf-go](#)

