

SIG-CLI - Open Doors

Maciej Szulik, Red Hat (@soltys)



KubeCon



CloudNativeCon

Europe 2020

Virtual

Mission



KubeCon



CloudNativeCon

Europe 2020

Virtual

kubectl

k8s.io/cli-runtime, k8s.io/sample-cli-plugin

kustomize

sigs.k8s.io/cli-utils

krew

sigs.k8s.io/krew-index

kui





Tech leads:

Maciej Szulik (@soltys), Red Hat

Phillip Wittrock (@pwittrock), Apple

Chairs:

Maciej Szulik: (@soltys), Red Hat

Sean Sullivan (@seans3), Google



SIG meeting:

Biweekly on Wednesdays

at 06:00 CEST / 12:00 EDT / 09:00 PT

SIG CLI Slack Channel:

[#sig-cli](#)

Email group:

kubernetes-sig-cli@googlegroups.com



KubeCon



CloudNativeCon

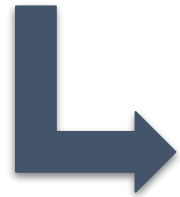
Europe 2020

Virtual

What is this **staging**
thing everyone keep
talking about?!

What is staging?

k8s.io/kubernetes



[staging/](https://k8s.io/staging/)



[api/](https://k8s.io/api/)
[client-go/](https://k8s.io/client-go/)
[cli-runtime/](https://k8s.io/cli-runtime/)
[kubectl/](https://k8s.io/kubectl/)
...

 publishing bot



k8s.io/api
k8s.io/client-go
k8s.io/cli-runtime
k8s.io/kubectl
...

kubectl is in staging - partially



k8s.io/kubernetes/pkg/kubectl/cmd



master [kubernetes](#) / [pkg](#) / [kubectl](#) / [cmd](#) /

Go to file

Add file ▾



knight42 refactor: move kubectl-cp to staging ...

✓ 1b32dfd on Jun 16 [History](#)

..

auth	Run hack/update-vendor.sh	2 months ago
convert	Run hack/update-vendor.sh	2 months ago
plugin/testdata	Update test data to reflect code move to staging	12 months ago
BUILD	refactor: move kubectl-cp to staging	last month
cmd.go	refactor: move kubectl-cp to staging	last month
cmd_test.go	Update test data to reflect code move to staging	12 months ago
profiling.go	Unexport kubectl cmd profiling	12 months ago



KubeCon



CloudNativeCon

Europe 2020

Virtual

Why kubectl is using outdated kustomize?!

<https://github.com/kubernetes/kubectl/issues/818>

<https://github.com/kubernetes/kubernetes/issues/82905>



KubeCon



CloudNativeCon

Europe 2020

Virtual

How a
kubectl command
is structured?!

Sample command



type `<command>Options struct`

holds all flags and necessary helpers for command to work

func `(o *<command>Options) Complete(...) error`

fills in all the necessary bits for the command to work

func `(o *<command>Options) Validate() error`

validates values in Options structure

func `(o *<command>Options) Run() error`

runs the actual logic

Sample command



k8s.io/kubectl/pkg/cmd/annotate/annotate.go

```
45 // AnnotateOptions have the data required to perform the annotate operation
46 type AnnotateOptions struct {
47     PrintFlags *genericclioptions.PrintFlags
48     PrintObj   printers.ResourcePrinterFunc
49
50     // Filename options
51     resource.FilenameOptions
52     RecordFlags *genericclioptions.RecordFlags
53
54     // Common user flags
55     overwrite    bool
56     list         bool
57     local        bool
58     dryRunStrategy cmdutil.DryRunStrategy
59     dryRunVerifier *resource.DryRunVerifier
60     fieldManager  string
61     all           bool
62     resourceVersion string
63     selector     string
64     fieldSelector string
65     outputFormat string
66
67     // results of arg parsing
68     resources []string
69     newAnnotations map[string]string
70     removeAnnotations []string
71     Recorder genericclioptions.Recorder
72     namespace string
73     enforceNamespace bool
74     builder *resource.Builder
75     unstructuredClientForMapping func(mapping *meta.RESTMapping) (resource.RESTClient, error)
76
77     genericclioptions.IOStreams
78 }
```

Sample command



k8s.io/kubectl/pkg/cmd/annotate/annotate.go

```
115 // NewAnnotateOptions creates the options for annotate
116 func NewAnnotateOptions(ioStreams genericclioptions.IOStreams) *AnnotateOptions {
117     return &AnnotateOptions{
118         PrintFlags: genericclioptions.NewPrintFlags("annotated").WithTypeSetter(scheme.Scheme),
119
120         RecordFlags: genericclioptions.NewRecordFlags(),
121         Recorder:    genericclioptions.NoopRecorder{},
122         IOStreams:  ioStreams,
123     }
124 }
```

Sample command



k8s.io/kubectl/pkg/cmd/annotate/annotate.go

```
126 // NewCmdAnnotate creates the `annotate` command
127 func NewCmdAnnotate(parent string, f cmdutil.Factory, ioStreams genericclioptions.IOStreams) *cobra.Command {
128     o := NewAnnotateOptions(ioStreams)
129
130     cmd := &cobra.Command{
131         Use:                "annotate [--overwrite] (-f FILENAME | TYPE NAME) KEY_1=VAL_1 ... KEY_N=VAL_N [--resource-version=version]",
132         DisableFlagsInUseLine: true,
133         Short:              i18n.T("Update the annotations on a resource"),
134         Long:               annotateLong + "\n\n" + cmdutil.SuggestAPIResources(parent),
135         Example:            annotateExample,
136         Run: func(cmd *cobra.Command, args []string) {
137             cmdutil.CheckErr(o.Complete(f, cmd, args))
138             cmdutil.CheckErr(o.Validate())
139             cmdutil.CheckErr(o.RunAnnotate())
140         },
141     }
142
143     // bind flag structs
144     o.RecordFlags.AddFlags(cmd)
145     o.PrintFlags.AddFlags(cmd)
146
147     cmd.Flags().BoolVar(&o.overwrite, "overwrite", o.overwrite, "If true, allow annotations to be overwritten, otherwise reject annotation updates that over")
148     cmd.Flags().BoolVar(&o.list, "list", o.list, "If true, display the annotations for a given resource.")
149     cmd.Flags().BoolVar(&o.local, "local", o.local, "If true, annotation will NOT contact api-server but run locally.")
150     cmd.Flags().StringVarP(&o.selector, "selector", "l", o.selector, "Selector (label query) to filter on, not including uninitialized ones, supports '=', ':",
151     cmd.Flags().StringVar(&o.fieldSelector, "field-selector", o.fieldSelector, "Selector (field query) to filter on, supports '=', '==', and '!='.(e.g. --fi")
152     cmd.Flags().BoolVar(&o.all, "all", o.all, "Select all resources, including uninitialized ones, in the namespace of the specified resource types.")
153     cmd.Flags().StringVar(&o.resourceVersion, "resource-version", o.resourceVersion, i18n.T("If non-empty, the annotation update will only succeed if this is")
154     usage := "identifying the resource to update the annotation"
155     cmdutil.AddFilenameOptionFlags(cmd, &o.FilenameOptions, usage)
156     cmdutil.AddDryRunFlag(cmd)
157     cmdutil.AddFieldManagerFlagVar(cmd, &o.fieldManager, "kubectl-annotate")
158
159     return cmd
160 }
```

Sample command



KubeCon



CloudNativeCon

Europe 2020



k8s.io/kubectl/pkg/cmd/annotate/annotate.go

```
162 // Complete adapts from the command line args and factory to the data required.
163 func (o *AnnotateOptions) Complete(f cmdutil.Factory, cmd *cobra.Command, args []string) error {
164     var err error
165
166     o.RecordFlags.Complete(cmd)
167     o.Recorder, err = o.RecordFlags.ToRecorder()
168     if err != nil {
169         return err
170     }
171
172     o.outputFormat = cmdutil.GetFlagString(cmd, "output")
173     o.dryRunStrategy, err = cmdutil.GetDryRunStrategy(cmd)
174     if err != nil {
175         return err
176     }
177     dynamicClient, err := f.DynamicClient()
178     if err != nil {
179         return err
180     }
181     discoveryClient, err := f.ToDiscoveryClient()
182     if err != nil {
183         return err
184     }
185     o.dryRunVerifier = resource.NewDryRunVerifier(dynamicClient, discoveryClient)
186
187     cmdutil.PrintFlagsWithDryRunStrategy(o.PrintFlags, o.dryRunStrategy)
188     printer, err := o.PrintFlags.ToPrinter()
189     if err != nil {
190         return err
191     }
192     o.PrintObj = func(obj runtime.Object, out io.Writer) error {
193         return printer.PrintObj(obj, out)
194     }
```


Sample command



KubeCon



CloudNativeCon

Europe 2020



k8s.io/kubectl/pkg/cmd/annotate/annotate.go

```
222 // Validate checks to the AnnotateOptions to see if there is sufficient information run the command.
223 func (o AnnotateOptions) Validate() error {
224     if o.all && len(o.selector) > 0 {
225         return fmt.Errorf("cannot set --all and --selector at the same time")
226     }
227     if o.all && len(o.fieldSelector) > 0 {
228         return fmt.Errorf("cannot set --all and --field-selector at the same time")
229     }
230     if !o.local {
231         if len(o.resources) < 1 && cmdutil.IsFilenameSliceEmpty(o.Filenames, o.Kustomize) {
232             return fmt.Errorf("one or more resources must be specified as <resource> <name> or <resource>/<name>")
233         }
234     } else {
235         if o.dryRunStrategy == cmdutil.DryRunServer {
236             return fmt.Errorf("cannot specify --local and --dry-run=server - did you mean --dry-run=client?")
237         }
238         if len(o.resources) > 0 {
239             return fmt.Errorf("can only use local files by -f rsrc.yaml or --filename=rsrc.json when --local=true is set")
240         }
241         if cmdutil.IsFilenameSliceEmpty(o.Filenames, o.Kustomize) {
242             return fmt.Errorf("one or more files must be specified as -f rsrc.yaml or --filename=rsrc.json")
243         }
244     }
245     if len(o.newAnnotations) < 1 && len(o.removeAnnotations) < 1 && !o.list {
246         return fmt.Errorf("at least one annotation update is required")
247     }
248     return validateAnnotations(o.removeAnnotations, o.newAnnotations)
249 }
```

Sample command



KubeCon



CloudNativeCon

Europe 2020



k8s.io/kubectl/pkg/cmd/annotate/annotate.go

```
251 // RunAnnotate does the work
252 func (o AnnotateOptions) RunAnnotate() error {
253     b := o.builder.
254         Unstructured().
255         LocalParam(o.local).
256         ContinueOnError().
257         NamespaceParam(o.namespace).DefaultNamespace().
258         FilenameParam(o.enforceNamespace, &o.FilenameOptions).
259         Flatten()
260
261     if !o.local {
262         b = b.LabelSelectorParam(o.selector).
263             FieldSelectorParam(o.fieldSelector).
264             ResourceTypeOrNameArgs(o.all, o.resources...).
265             Latest()
266     }
267
268     r := b.Do()
269     if err := r.Err(); err != nil {
270         return err
271     }
272
273     var singleItemImpliedResource bool
274     r.IntoSingleItemImplied(&singleItemImpliedResource)
275
276     // only apply resource version locking on a single resource.
277     // we must perform this check after o.builder.Do() as
278     // []o.resources can not accurately return the proper number
279     // of resources when they are not passed in "resource/name" format.
280     if !singleItemImpliedResource && len(o.resourceVersion) > 0 {
281         return fmt.Errorf("--resource-version may only be used with a single resource")
282     }
283
284     return r.Visit(func(info *resource.Info, err error) error {
285         if err != nil {
286             return err
287         }
288
289         var outputObj runtime.Object
290         obj := info.Object
291     })

```


New contributors / first time issues



[Remove generators from kubectl commands #93100](#)

sync with [@zhouya0](#) and [@aubm](#)

[kubectl --local is not quite local ... #93188](#)

sync with [@solysh](#)

SIG-CLI bug scrub every 4 weeks on Wednesday
next August 19th at 06:00 CEST / 12:00 EDT / 09:00 PT

More soon to be announced at

kubernetes-sig-cli@googlegroups.com



SIG meeting:

Biweekly on Wednesdays

at 06:00 CEST / 12:00 EDT / 09:00 PT

SIG CLI Slack Channel:

[#sig-cli](#)

Email group:

kubernetes-sig-cli@googlegroups.com