<https://medium.com/better-programming/k8s-tips-using-a-serviceaccount-801c433d0023>

<https://www.youtube.com/watch?v=KBTXBUVNF2I&ab_channel=CNCF%5BCloudNativeComputingFoundation%5D>

<https://itnext.io/deep-dive-into-how-kubernetes-rest-api-works-517c86f1640b>

cloudaark – good

<https://medium.com/@cloudark>

**Tutorial: Deep Dive into the Operator Framework for... Melvin Hillsman, Michael Hrivnak, & Matt Dorn**

<https://www.youtube.com/watch?v=8_DaCcRMp5I&ab_channel=CNCF%5BCloudNativeComputingFoundation%5D>

Focus on bold highlights.

**oc explan**

**kubectl explain replicaset**

**you should be using that every day right they explain command actually coop CTL explain**

**deployment COO CTL explain replica set that thing talks to the API right and**

**get you back information on how you can actually fill your declaration right out**

so in this manifest we have here the way to think about this when you look at this in the future think about it as

**four different parts you're looking at four different parts**

**First part** is the GVK or a group version kind. We also call that the type meta that's the group version and kind here

**Second** metadata or object meta you know about this right you can put a namespace in there it's your if you're not scoped

to a specific namespace when using cube CTL it will our view our scope to a specific namespace when using cube CTL

you don't have to put a **namespace** in their **annotations** go in there the **owner references** end up in there which we'll talk about later **labels** we can duplicate though we can put a lot of stuff in there okay

**Third spec** what about spec so in our spec here this is our declaration this is what we want right so what we're actually doing is we're saying I want to have make this replica set that has five replicas right and then we can specify one container or

multiple containers this is yourdeclaration this is what you are requesting right this is yourdeclaration what you want now the

**Fourth** section here is status it seems like you know a lot of people that get new to kubernetes are kind of like well I

assumed when we talk about operators as kind of comparing the actual state of the cluster to the desired State this

being the desired state that this is the actual state a lot of people think that but no the status is only written by

your controller and it's a way to provide information to your users it can also if you're creating maybe a more

advanced operator be used as a way to store previous old state so if you have to you want to check and only do

something and this is in ready status your operator can actually look at that and decide whether or not to proceed if

that's in a certain status right but **my point about talking about this is that this is something that a user does not fill out that a human does not fill out your controller does it to provide information to you the human or perhaps other operators running** in their cluster

**Operators take advantage of custom resource definitions**

so those who have used operators before you know that these things are incredible right **because they allow you to extend the API via the API okay so you don't have to actually go and grab the kubernetes code and then like for example if you want it to add a new resource** like let's say you want it to add a new resource called coop con so you could do like kubectl get coop con or something right or have an end point if you're visualizing the API endpoint right you're visualizing like B 1 beta 1 slash like namespaces slash whatever slash coop con like you've the actual resource you don't have to actually bring down the code to make that change you send an API call to modify and create that new resource on the API that's pretty powerful command right because I mean it literally impacts the entire cluster and everybody

using the entire cluster

just so you can see what one looks like this one is two called kinder fauna what do you think this does if you had a guest you had to take a wild guess what does this probably do if you were apply this right

it creates a group on an instance of something right this is just an example of showing you this you could see though

there's some configuration parameters in here right so it's kind of like rather than using like a configuration file or

a config map and having to mount that config map as like a volume mount right it's like a deployment or whatever like

we're just specifying what we want as a declaration and the spec section here right and there were cool thing about

creating si RDS or a custom resource is as an author I can actually design this I can design what users can put into

here right which is really really neat

so here we have an example custom resource



so let's go ahead and just show you an example of let's extending the kubernetes api by creating our very own object or resource via CR DS so for those of you who've done this before just go play on your play on the lab environment and look around and you know probably boring you you've done this but if you're never if you're not really

familiar with CR DS and you've been in the conference for the past few days and you're like what is this thing CR D thing anyways this is basically what it is really straight forward creating the CR D **so what we do here is we create the CR D first the custom resource definition and literally the kind is custom resource definition** this is the first step to creating a CR D you can see that this right here is the customized piece my sequel's I'm just making that up **I'm going to create an operator that let's say creates my sequel** easier said than done as we're gonna see but point is as I'm creating my custom name here and **my custom group**

here in here what's the group the **group is nothing more than a way to disambiguate different kinds that may have or let's imagine that they're in companies all come out with my sequel operators right and they probably all create the same kind my sequel right**

**well how do you tell the difference one's gonna be company a one's gonna be got company B** right so that B company a Company B and then here we're saying here the version well this one starts with v1 I should probably fix this slide because what do we want to start with usually if you want to start with V 1 alpha 1 right that's what you want to start with usually so I should probably fix that so don't look at that right now ok

**scope namespace scope namespace** this is saying after I create this custom resource deficit definition which by the way let me ask you a question what do you what do you think that this object here is gonna be scoped to first of all do you think you need privileges to do this would you think you need pretty wide privileges to create

something like this **you're modifying the kubernetes api think about that yes you need some pretty cluster wide privileges to do this right you're literally modifying the API but then this block here says scope namespace** why are we talking about scope namespace right **namespaces are how we do multi-tenancy and Kubernetes** because after we create this and we want

to **create the kind my sequel object** right we're going to want that scoped to a namespace so if we have example my

sequel that's gonna dis a namespace one but if I want to create another example my sequel that's the name literally we

know that we can't have multiple objects with the same name right so we gonna put that into another namespace so that

means it's gonna be scoped to the namespace **what do you think the opposite option of this is if you had a guess cluster wide** the exactly right very good yep so this is just a couple ways to refer to it and **some aliases** that we can refer to which is very awesome I can say **kubectl get ms** and it understands what I'm talking about **how does it do that the client does discovery right right** when you type the word it's gonna scan the API for the alias or nickname and it puts that in the dot what is it what's the directory do all those with a dot cache or dot yeah the coop needs a coop CTL is it a dot cache directory that it puts it in I think it's a **dot cache directory** or something right

and then you create the CRD after we verify the creation of the CRD we then go in and we say oc get CR D and how do we know it's there is it there yes it's there we say oh so you're wondering probably wondering what's OCokay so OC is the OpenShift client all

right maybe I'm a little biased we originally an open ship it's just like coop CTL what is it what is what do these things do behind the scenes you're just an API calls right they're just talking to the communities API so **OC is just like coop TTL OC just has some extra open shifty stuff added to it** but it pretty much acts like coop CTL same thing you can use both interchangeably

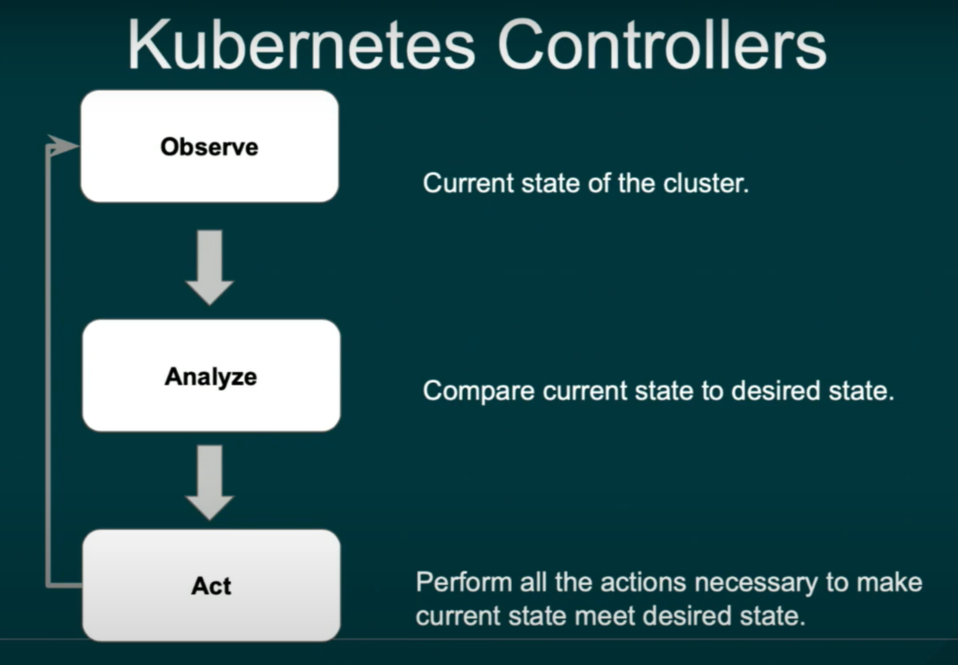
okay work really great so we're saying coop CTL get CRD and it comes back and

goes that's your CRD let's now actuallyverify our new my sequel resource and object so what we're gonna do now is wehave to see our D created check this outnow I can actually go up here and say OCor cube CTL get my sequel

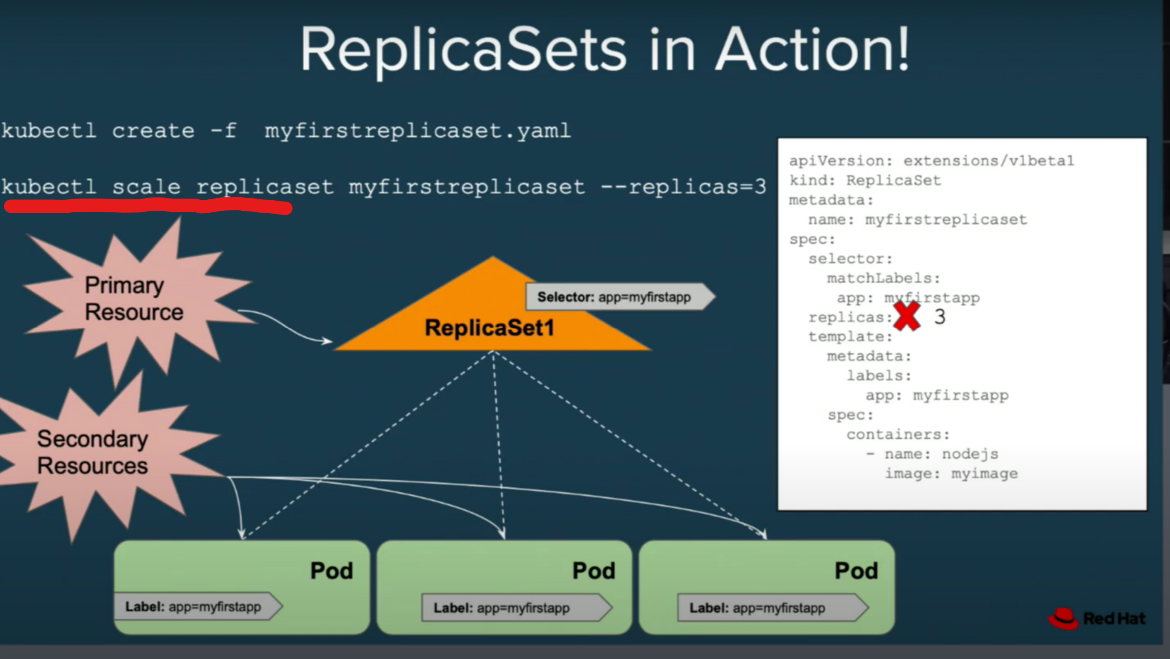
I can actuallysay get my sequel and what happens is itcomes back and says no resources familyeverybody thinks at this point when theysee this they're like oh it didn't workbut **this is a good thingthis means that behind the scenes coopCTL or OC talked out to the API andactually found an end point when I sayend point it found a Ning spaces /v 1alpha 1 slash whatever right it's aslash my sequel it's bound** it but guesswhat there were no objects in there yetwe haven't created one yet so let's goahead and create an object so when wecreate this object it's gonna look likethis DP'd example.com class v1 it has mysequel in it it says name WordPress userpassword you may be thinking like wheredid you get this stuff from how did youhow did you know to put this stuff inthere well let's just say that you canactually customize your custom resourcedefinition with the same information youget back when you use OC or coop CTLexplained so for those you who use theexplain command you can inside your CRDyou can actually program or customizewhat comes back so I could actually haveit's a coop CTL explain my Segal and itwill tell me okay you can put youhere you can put a password here you canput you know food here or whateverit will give you and you can program itto give you instruct instructions butthe reality is that this right here youknow this is not really doing much yetwe're just adding this to the API that'sit you can probably guess what we'redoing here we're trying to create awordpress database right this is anexample so we go ahead and create thatobject after we create that object wesay coop CTO get my sequel it comes backit says WordPress if I want to actuallylook at the object my object is actuallyin here it looks good it's persisted onthe APIthus persisted where I see it where isit actual where is it where is itsitting where's the source that'sexactly right it's sitting and EDD rightthe greatest key value store

okay sothat doesn't really do much for me whenit comes to an operator it gets me done with number one which is number one what's number one required for anoperator I need a customer resourcedefinition right yeah that's right so itgets me that that's good but a customresource it needs a controller to act upon its presence so without if controller you just have this thing onthe API and that's it and that's notgonna get you much right so what docontrollers do **controllers literally sitin the cluster they sit in the clusterand they just watch the API now how they watch the API** we usually talk about we run a version of this coarsely over acouple days and you get into thehardcore specific sorta like it formershared informers to watch all that goodstuff that is you who use client go knowand love I know you all know what I'mtalking about but those years ago nowjust know it watches the API okay itjust watches the API for the currentstate of the cluster then **after itwatches the current state of the clusterit analyzes okay it compares the currentstate of the cluster like how many podsare running and how many config maps andwhatever I want to look for is thecurrent state to the desired state wheredoes the desired state live you tell meyeah that's exactly right it's in thecustom resource in which section thespec section that's your declaration**right that's what you want that's whatyou want in the declaration so itcompares what is do I have any my sequelthings runhow many pods whatever do I meandeployments deployments so I have anyyou know Peavey's whatever to what Iwant in my spec **and if I don't havewhat's desired what does the controllerdo it acts right it performs all theactions necessary to make a currentstate me the desired state that's whatit does** and that loop runs whenever itreceives an event and guess what forthose you use kubernetes and you'venever used it internally which istotally fine maybe that's not your scopeof your your you know what you do day today it's totally possible to usecurators every day and not to get intothe internals just not part of the scopeof what you do day to day but this thingright here we talk about events we'renot

talking about cube CTL get eventsthat's not what we're talking aboutwe're not talking about coops to killgood



**events we're talking about eventsthat you see you get when you watch theAPI you get a add event when it'ssomething when that CR is created youget a update event when the update ismodified I'm sorry when the CR ismodified** how do you modify a CR you canrun coops ETL replace right on thecommand mockcommand line how else how about **some ofyou probably use coops ETL edit rightyou do kubectl edit and you go inthere and you like using your editor ofchoice** and editing it that way maybe youdo it that way that's an update and thenif you delete it **so this thing onlyhappens this trigger only happens forthose three different events add updateand delete**



here replica sets inaction remember I told you thatunderstanding what how awful the setswork can really help you inunderstanding operators here's ourreplica set which notice how it has aselector it's saying I'm looking for a Pequals my first app we have a pod hereapp equals my first app that's how itknows to pay attention to this pod ithas the same labelwe have replicas set to one what happenshere we change replicas to three andit's smart enough to know to spin up twomore okaypretty straightforward

that's ourprimary resource these are our secondaryresources there's just a nickname thatwe use to refer to something like thisso

imagine the replica set as sort oflike the custom resource think of itlike that kind of like the customresource right and these are the kind ofthe children or secondary resources thatare being controlled by that thing bythat replica set

now **owner references** are things that you may not know you maynot know what an Oda reference is rightyou mean to have no idea what areference is but you use them all thetime if you're spinning up deploymentsyou may name never just see maybe thisis really notice **owner references helpus with garbage collection** right can youthink about it you're creating thesethings called deployments or you'recreating a custom resource and it'screating things like you don't want itwhen you delete the custom resource oryou delete the deployment or whatever oryou delete the replica set in this caseyou don't want your pods hanging outright after the things been deleted youwant those to go away toothat's what owner references do okay soif we have a replica set we spin up areplica set and we have pods here wewant three replicas they all have anowner reference now what the ownerreference does for those who don't know if the owner goes away so in this casethe owner here is this replica set ifthe owner goes away and gets destroyedand never comes back well **I just say ifit goes away guess what happens to theorphan here it goes away right kind ofsad right so pod has a parent parentgoes away gone guess whatchild is gone that's how they work whocontrols that logic the garbagecollector that runs and is embedded intothe controller manager in kubernetes** sothis is embed it you get this for freein kubernetes it's not an operator thingokay it's not an operator thing **veryimportant when you're creating anoperator understanding how ownerreferences work** because you don't wantto have people you don't want to leave amess right you don't want to leave amess when someone deletes your customresource

okay so how does the replicaset control loop works it gets triggeredthis iswe showed you before itfinds pods matching the label selectorwe compare the matched versus desiredpod count right so think about this ifthere's too few pods when we'recomparing the match versus desired rightwe're gonna create some additional podswe'll have some logic in there ifthere's too many what are we gonna dodelete them right we're gonna deletethemget rid of them we think and guess whatthis **thing gets triggered anytime therewas an event okay** and then you canactually see these in action here it's alittle getting a little more detailed

ou have a cube API you have a replicaset controller the replica setcontroller watches for replica sets soit sees you bed space up replica setsand you have pods it also watches forpods but not just any old pods itwatches for pods that are owned by thereplica set and check this out create areplica set there's an event that getstriggered we then do a list against theAPI for the actual state turns out thatthe actual state is zero it's less thanthe number of desired replicas here sowhat do we do we then create one podguess what we get an event based off thecreation of that one pod we then listagain get the actual state now we haveone all right you all seeing how it'splaying on here right and this is thetriggering here everything you see theseevents and I think it's important foryou to understand that these events yourcontroller doesn't actually know thatthey're at events it just knows it thereyou get an add event sure but yourcontroller isn't care if it's an ad ofend of the care with an update eventit's something you have a bunch of ifstatements in there in your logic rightit's not like that we're gonna look atthe logic coming up here and some you'revery fast because I don't have a lot oftime oh now we're happy why are we happybecause now three is equal desired isequal to what actual and that's good andthen guess what when that happens yourcontroller stops

every time you see an event that is thecontroller what we call it was anickname for it reconciling that this isthe control on the right is thecontroller on the right is thecontroller right so just takeyeahyou you could said he's asking if youwant three pods you might code it toloop three times or something alongthose lines or the 301 burst you havethree in one burst yeah oh three all inone go is that it yeah and sometimes youdo want to do that but sometimes it'ssufficient and even desirable to justtake one step in the direction of thedesired state instead of going all theway to the desired state it reallydepends on what exactly you're deployingand and why in the case of pods probablyit would be fine to go ahead and deployall of them like figure out the entiredifference between desired and actualand just do it all in one shot but inother cases it's advantageous to notyou're gonna come around for anotherreconcile loop immediately anyway sothere's really no harm and in some waysit's a better pattern to do itincrementally it's easier to test andthat's reducing yeah and the performanceis obviously doing it's very fast rightwhen you do it this way as well okay soand look what happens when we delete apod we get a bleed event too right sothere goes our delete event and thenwe're gonna have you that actual stateagain which says there's two so Iobviously have to go and create a newpod we get a pot event and then we'rehappy okay

so going back to the mysequel example check it out so rememberwe didn't create a my sequel controlleryet we were to show you how to replicatebut notice how notice how the the the CRis sitting on the API here by itselfnothing is happening in our cluster butwe don't have a controller so what wehave to do is that we actually have togo and very quiet we have to go anddeploy a controller in our cluster rightand notice how we have these controllersnow this is actually I got to changethis slide because even though **we  
wantto have multiple controllers runningguess what only one is allowed toactually contact the API and act on thatobject you don't want these controllers stepping on each other's toes here rightso we have in place leader election soone controller will be able to actuallywatch the API here** so don't be confusedby this I need to fix the slide andthat's gonna dustor my seagull pod but obviously I'mmissing a lot of stuff here if I want tohave multiple my seagull pods youprobably want to have these deployed asa staple set we probably want to havemultiple TVs deployed didn't have tinybut all those graphics on there buthere's reality it's actually morecomplicated than thatI mean you're free to use any of theprimitives or any of the artifacts youget with kubernetes to actually go anddeploy this my sequel instance howeveryou want to do it right you're free todo that but it's really a lot morecomplicated than just creating somethingor reading it and updating or deletingokay think about

anyone here my sequeladmin think about all the things thatyou have to do when you manage my sequelright like all the tasks these are thethings that you need to think about whenyou build your operator these are thethings that **you're gonna codify and putinto your operator now how you put thosethings in there you can use the go youcan use go to do it** and if you'recomfortable go and that's where you feelcomfortable you can do that **but thinkabout this you can use ansible to create an operator to with the operator SDK and you're probably thinking to yourself huhI can use ansible to do create a mysequel operator** Matthieu by you when areyou pulling down there's my sequelansible modules right maybe that youthose either use ansible right so we'lltalk more about that we get to date theAngelo piece as well **but point is isthat this is the human operationalknowledge** we're talking about how youcodify that and how you put this in toyour operator and again we'll thinkabout more of an app think about an appthat you know and love this is kind ofgeneral my sequel the ins and outs ofthat app that's what you're gonna putinto your operator how you do that we'regonna talk about coming up

so to recapwe custom resource definitions pluscustom controller plus your knowledge equal soperators

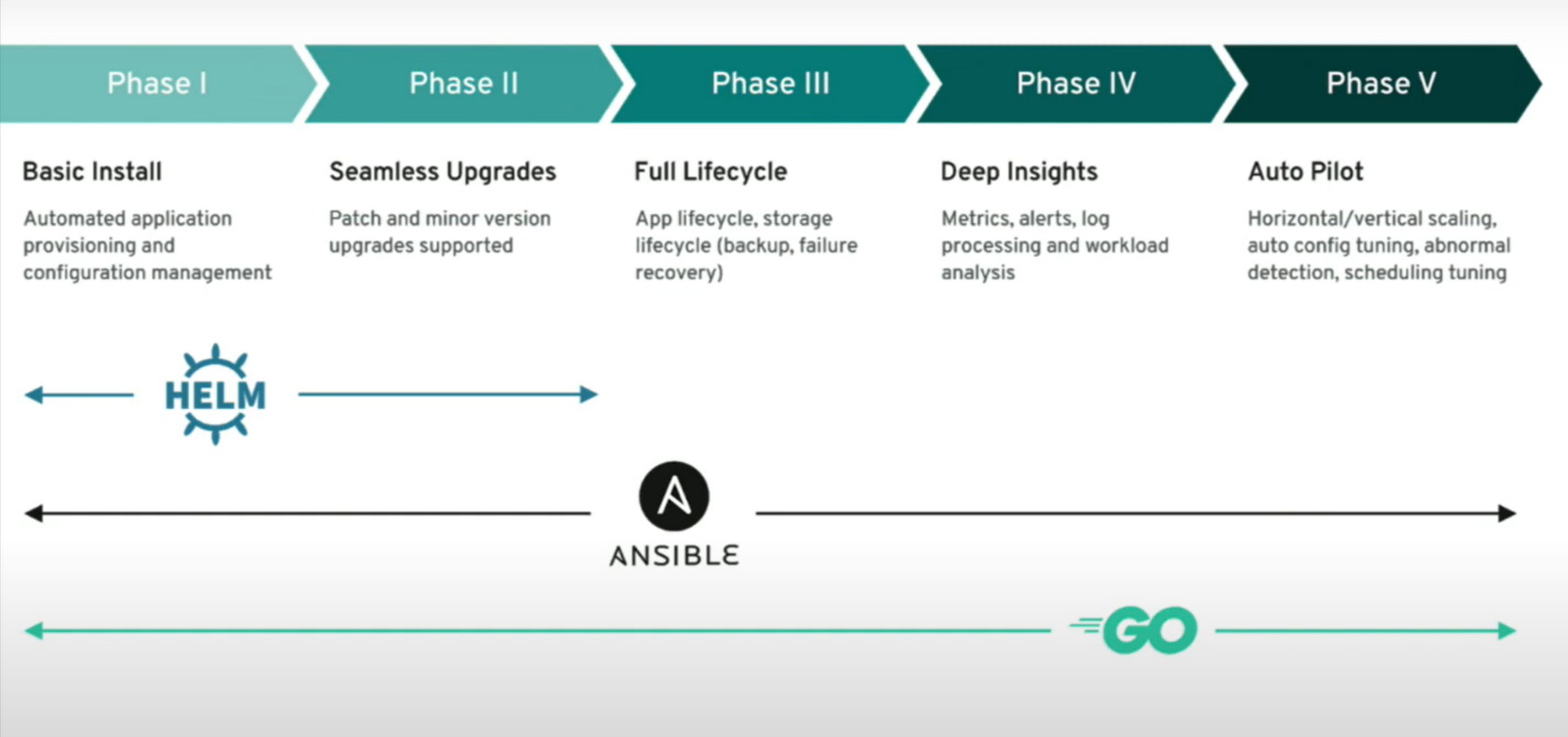
why they matter here's thewhole idea we're trying to build an as aservice experience right what we didn't buy that you go to AWS you go GCP you goto a sure you know you want to go youknow go ahead and provision a databaseor something is very easy you just youknow fill out some fields put some radiobuttons if you use the UI you knowyou're probably using terraform orsomething you know to actuallyprogrammatically interact with thatinfrastructure but you get what I'msaying it's like a very easy experienceyou build an ecosystem of software onkubernetes that can be as easy and safetoo **reliable to use as a cloud service one touch remotely managed one-click updatesright giving you that really nice youknow low touch experience an example ofan operate** on the dashboard lookssomething like thiswhen I say dashboards I think in whatdashboard what do you mean I just meanfrom the UI if you were gonna deploy anoperator like this is what we're goingfor like making it look like justclicking a button and going enable ithere's the sed operator that's whereyour apps running on kubernetesPrometheus and for monitoring and voltfor a secret storage

how do you create your ownoperator light before the operator SDKit's really for those you create anoperator you know can be a littlecomplicated right it's not just assimple as creating some resources andthen defining a controller with somepseudo look code that looks like thisright it's actually a lot morecomplicatedthey require many building blocks ofboilerplate code you have to researchand download tools to interact with theAPI first of all you start at the clientlibraries for those you who know there'snot a lot there's not a whole lot ofDoc's on these client libraries rightmaybe some of you who or you know maybeyou may not be you doesn't matter youjust read the code anyway but you knowgo Python Java dotnet what is this stufffor it's for programmatically working inthe language that you're mostcomfortable in to interact with API weinteract with Guber Nettie's rightspecifically the communities API

yepokay well I didn't realize thatknowledge of Informer's sharedInformer's or object cache and eventhandling why do we care about objectcache and event handling think aboutthis what do controllers do they watchthe API you have a whole bunch ofcontrollers running in the cluster righttons of controllers running in theclusterthey're always wedding in the clusterthere watching the API what does thatmean for the API that **poor API isgetting beaten** right so how do we makeit more efficient the controller has youknow caching we have a work queue wherethey work off of communicating desiredstate you have to build that if you dothis from scratch it's really annoying

this is an example we're just usingclient code directly just showing youwhat a sample project would look likeand we need an **easier way to createo perators and easy way to manag eoperators that's what the operator framework comes in it** **SDK allows you consists of operator to create an operator with go or Ansible or helm you can convert all your existing helm charts to an operator if you want to okay we're not competingwith helm helm is awesome** but if for some reason you want to create anoperator and put it in the operator hub you can do that with or put it in youroperator hub which is nothing more thanlike an app store you can create yourown if you want kind of your ownregistry of operators

**MATURITY MODEL FOR HELM, ANSIBLE AND GO**



helm helps you thehelm helps you do that with the or thehelm side of the operator SDK this isthe full lifecycle going through fivestages of maturity and every we're up toautopilot which is fully running on itsown auto scaling responding to custommetrics for those you who are devsyou're developing operators Prometheusmetrics definitely learn how toinstrument your app that's reallyimportant using the previous librariesbecause autopilot responds based offthresholds of those custom metrics

okaythat's exactly right you can chooseeither one you can choose helm you canchoose Anne's floor you can choose goyeah that's exactly right that's whatthat's what it's saying heremaybe someone if you all work on hellmore if anyone here works in the homecommunity we can we can talk about thisbut that's what this is suggesting yeahand then

OLM **OLM handles theinstallation the upgrade did the wholedependency management with operators howdo you install them how do you upgradethem all that good stuffthis is actually giving you that appstore like experience** think about thiswhen it comes time as an operator authorfor you to roll out a new version ofyour operator to your audience kind oflike an app store oh I have to you knowone of you do you want version 2 what isit gonna do that's what LM does for youwe might have time to talk about thatlaterit's totally upstream fully availableoperator hub that IO this is these runin any kubernetes clusters it's kind oflike your App Store like experience

caddy coda housecat ACOTA doing is itworking oh well good we apparently hadso many connections so quickly to KaiDakota that we set off an alarm and itthought it was under attack and theywere all from the same IP so it said ohno no we're not let these people in sothe threshold has been increased weshould be in good shape anybody's stillhaving troublewell slow ok hopefully it'll still beworkable and I think we're about to diginto that now yeah we'll find out thanks



oh yeah by the way there's some bonusmaterials I forgot to check the websiteit should still be up workshop thatquarter less trained me a lot of slidesyou can download from there workshopthat core OS trained me trained me it'slike there's lots of good stuff on thereyou can download lots of resources lotsof access to the support Docs and thechat and all that good stuff please takea picture real quick I'm sorry forspeeding through all this stuff I toldMichael I promised Michael I said Iwon't make this one long and it ended uplong the intro long I have a hard timecondensing this is a big topic it's abig topic ok ok so by the way get yourif you want if you all working on anoperator you want to get it certifiedand featured in openshift so everyonesees it and even if you you know have aan enterprise offering or something thatthat you know so after people areevaluating the upstream version cometalk tothe people in the front here that's whatthey do they help people get theiroperators certified going through aseries of very strict testing on yourimage and your operator metadata

okay sowe're gonna go ahead and create anoperator with go and it's going to bevery quick and I'm gonna ask I'm nottrying to be rude but it looks likelet's hold the questionsI hate rushing through this type ofstuff I you know it's just but this iswhere we're condensed for time but stickaround afterwards we're gonna answer allyour questions again we're gonna be herefor you to help you so what we're goingis we are going to not the Catesfundamentals which is a really coolexercise that helps you explore the APIbut the **operator SDK with go pod set theoperator SDK with go pod set** and

**learn.openshift.com**

whatthis is now look here's a reality Inoticed with these workshops a lot ofpeople like just this is the quiet timewhere it's like go work on this but I'mjust gonna walk through it okay maybe ithelps okay so I'm just gonna walkthrough it very quickly you're free todo this on your own time oh yeah whenyou leave you can hit all theseexercises not add flash training butjust knock off the slash training rightand just go to learn dot open chip calmand you'll have all the modules on thereall the modules will be on there so whenyou leave learned open chip calm we'llhave all this stuff so it's not goingaway so I guess I have to wait for myenvironment to come up now I guess okaylet's do this

what we're gonna do hereis we are going to use the operator SDKcommand okay and we're gonna say hereoperator SDK new pod set - operator - -type equals go operator - SDK new and

now the gentleman in the back set wasasking do you have to use all thesethings how does it so no here we'resaying I want to create an operator withgo now you may be wondering should I gowith go or should I go advancable whichone do I use here it really depends itdepends on what you're comfortable withthat's all if you're an ansible shop andyou all you have tons of antal modulesthat you all have stored and tower oryou use you like to fetch them fromgalaxy or **however you want to you knowyouuse ansible you can use ansible tocreate an operator if you all are goshop you love go you code and go** you canuse go we're gonna create a go on herebut let's see if this one provision intime can anyone I don't know maybe if wewant to maybe your house yours yeah thisI thought we did this in the beginning Ithought that okay yeah okay oh here wego okay I'm good yeah okay alright goodyeah okay alright

so I'm gonna create anew project which for those you who arelike what's project are you talkingabout it's a namespace that's it openshape disease use the word projectthat's all it is

so I'm gonna click thisto create a new project and then I'mgonna create inside my little sandboxand my go path right this Red Hatdirectory and then I'm gonna navigate tothat directory and then I'm going tocreate the operator with go and I'mgonna say skip get in it because guesswhat we're not pushing this any we'renot pushing this to get so we don'tgoing to worry about

thatand there it goes now what's happeningberight now is you're getting all theboilerplate code and you can probablythose either use go you know what thatscreen looks like right third your thedepth screen that's going out andpulling down all those packages okaythat are being referenced in thatboilerplate code so getting all of thatall that go code from those importstatements that are in the top of thecode this is talking out to the net todo this so you do need a extra you knowyou need external connectivity to runthis

and so what's that you're probablythinking is this creating my operatorhow does it even know what kind ofoperator I want to create what'shappening right nowI'm confused this gentleman has aquestion yes

thank you okaywhat's the error was that okay oh okayoh yeah oh yeah by the way real quickyeah let me show you that somebody askedabout this where do though I know wejust want to show you it workshopwhat is it core OS train I have a minuteanyone for us trains I mean okay if youwant the slides click on support' andyou can download either of these linksthis was a from previous cube Khan andansible fest yes we did not update muchcontent here from the previous cube con

but we are totally revamping this forthe new version of OpenShift

of the URLworkshop core OS train dot me I'm gonnaput it on a big screen herethere we goThanks thank you

yeah so he's talking about thedifference between kubebuilder which isanother way to easily create operators

now with if you're all familiar with kubebuilder so they both **operator SDK and both kubebuilder both use somethingcalled controller runtime** right **controller runtime is a series ofpackages go packages that makes creatingoperators really really easy it wasdeveloped by the kubernetes authors**

actually or developers of the API andthere's some news that actually operatorSDK has been working with COO builderdirectly so are we gonna do we knowanything about the changes with theoperator SDK Michael and up and cuebuilder like as far as what's workingtogether still pending but we have weekly meetings with each other so weare not we're not that's they're **thes earen't two competing platforms so tospeak right**

**Ultimately the go features are gonna fold under kubebuilder and operator SDK will bring operator specific things on top of what key builder currently offers that'll be things like packaging for operator lifecycle manager perhaps testing and that sort of operator specific stuff that opera CK does today but that'll all be go focused and probably the ansible in home pieces willstay under the the operator CK umbrellaas it is now**

yeah so there's you're notthis is really a good you know which oneshould I go with they're basically gonnabe this have the same workflow operatorSDK is great if you want to be in theoperator lifecycle manager and get thosekind of features right building out allthe artifacts you need for OLMyeah we don't know yet but Oh Willadoesn't say this let's add is a goodquestion if I want to get startedcreating an operator today what do Ipick though right now tonight I want toget started with the operator and workthroughout the weekend which one am Igonna pick what should I do

**I would sayI mean today like right now tonightoperator CK still has the operator bitsthat are gonna get you all the way tothe finish line with operator hub forexample or does generally if you want topackage and distribute your operatoreven internally it's got that essentially that package managementfeature built into it and ready to go soI would do that today**

oh thanks yeahokay so okay I'm gonna go to this prettyquick so the first step to creating thisgo based operator is to add an API is toadd an API what does that mean to add anAPI? that means to create the customresource definition the custom webasically telling this this code this isthe custom resource definition that Iwant to use make the code aware of thatactual object that's what we're doinghereokay so we're saying operator SDK addAPI - - API version apt example.com /v 1alpha 1 so very creative name group hereapt at example.com and then we're gonnabe calling creating this **thing called apod set now** if you're wondering what isa pod set a pod set in this objectivethe kind of operator that we're actuallycreating does this right here it's verysimple **it's almost like a replica set and** that like you specify the number ofpods that you want right and then itgives you that number of pods and italso has the pod names in the statussection very very simple

maybe thinkingwell how is this different than areplica set it's really not muchdifferent at all just we're showing youhow this is a good place to start tounderstand creating a controller okayhere if I change the size to one thedeclaration the pod set should be smartenough to scale down so you want to havein the spec section replicas in thestatus section pod names okay

so just to give an idea what we're doinghere all right so we're gonna go andcreate this add API command

again you see what it's creating hereyou can see it's creating the CRD for us automatically it doesn't knowthe contents of our custom resource itdoesn't know what we want to have insidethe spec yet we have to tell it thatright

and never we want in this case wewant was at replicas that's what we oneside of respect right we haven't told itthat yet we're gonna have to tell itthat okay so now if you actually look atthe CR D the CR D has been created foryou alrightagain I'm skipping over so much usuallyyou know this is a lot of stuff andwe're skipping it over

now we're goinginto here is hopefully if this works foryouyou can drill down on the very top tothis file location pod set operatorpackage api's and i need to refresh itseems like the environments are workingagain for many people

so if you're stillstaring at that red error messagerefresh one more time hope it'll work

so what I'm doing here isthere we go okay all right so what I'mdoing here is I am going to create my spec I have to design my specright what do i wet in my spec I wantthe number of replicas so what I'm gonnado here is I'm gonna drill down and dothis types go file right right hereposit type let go this is where I designmy spec this is where I say these arethe fields I want right and this is whatI want them set to this is the status that I want

please open okay there we go all rightso what we're gonna do here is we'regonna say this is very easy okay katacode you have to do is click copy toeditor and it literally updates it justlike that but if you're wondering whatthis is spec section status section thisis a field that can be referenced and goit's gonna be a 32-bit integer and thenwe're this is a JSON tag it's basicallysaying inside the JSON in thatdeclaration we're gonna have replicas here and in the status section or havethis array of pod names that's all we'redoing and after you run that commandafter you update that code you run thisgenerate Kate's command for those of youthat are super controller savvy theseare just the code generators that arerunning in the background here you knowyou don't know what that is you neveruse controller don't worry about it butjust understand that just a littlecommand that we did to run those codegenerators to create the type objectsbased off this information here and

generate open API you all have you knowit's a really cool thing about this isis you all when you all do coop CTO editlike an object you know when you go inthere and you try to change somethingand you hit return I'll satisfy that youmade your change and it like comes backat you it's like you can't do thatthat's validation that's validationtaking place right you can do that inyour spec so you can say users can't putfour replicas you know some like longstring or something they can only put it32-bit integer e you can say they canonly put an integer between 1 & 5 youcan put all that information into yourvalidation and that validation goes intothe CRD

and now we have our C Rdcreated and there it is pod sets tappedat example.com now comes the really coolpart where we have the logic controllerthe controller is the thing that doesthe reconciling remember that screenthat we showed youwith the back-and-forth the the kind ofthe looping this is the controller thatdoes that reconciling

and then now whatwe're gonna do is we're gonna go andupdate our logic let's see let's go intoour controller pod set pod setcontroller okay there we go you get somedefault logic but it's very very basicit just creates one pod

we're gonnacreate them this is what our logic lookslike okay look tipless and i promised wewould typically go through all this andtell you everything that's going on butwe don't have time we got to talk aboutHanceville operators

so please come Iwant you to scan this I want you to getyour questions ready and come see usafterwards and ask his questions aboutthis okay but the most important logicif you're kind of wondering what'sreally going on herethe real meat of it if you will like theactual thing that we should focus on isif the number available is greater thanthe this is the comparison right thenumber available is the actual statethis is our declaration if this isgreater than this we're gonna go aheadand scale down if this is less than theDeclaration we're gonna scale up that'sthe most important piece in all this right so I'm gonna go ahead and applythis and we should be updated there you don'thave to click save or anything it shouldsave automatically and

**now we're gonna run this controller using the up local command you may be thinking I don'tunderstand why aren't you deploying thiswith a deployment like into the clusterit's because the up local command allowsyou to compile this code and talk toAnne Cooper IDs cluster if it's externally accessible so you're literally running a controller outside of the cluster** and it can

**Process of deploying controller into the cluster:**

**actually think about this the reason why we have this command and we made this command is because can you imagine if every timeyou made a little code change you'd haveto literally compile your image push itup to a container registry go and you know put that the name into a deployment**

imagine how annoying that workflow wouldbe so this is a really nice workflow tojust instantly connect to your clusterand try your code against that clusterok and it's running ok how do I knowit's running well

I have a my screenhere and let's go ahead and make sure itworks what I'm gonna do here to makesure it works is I'm gonna open up a newterminal and I'm going to go into thisdirectory here and I'm gonna notice that this is set to size remember we set itto replicas so how do we get this toactually work but step the replicas whatwe're gonna do is we're just gonna openup our deploy folder go to see our DS goto see our y amyl and click our magicalcopy to editor buttonisn't this great wouldn't it be awesomeif Cody was actually like those you justlike clean fun no that wouldn't be funall right it would be very fun all right

so then we then we actually go and wesay OC create - F deploy CR DS let's go ahead and create thatand then we say OC get pods and thecontrol or check this out this is prettycool if you look at the terminal noticehow instantly when I created that customresource my controller got an event youcan look at all the logs here and thenit scaled up based off that event etcetc you might be thinking I don't eversay this is such a simple example whileyou do it showing us it's like a replicaset was a big deal it's academic or justyou study thisand this will prime you for understanding how op the controller orloop works play with this play with thisplay around with it modify it break itget comfortable with how this controlleris actually working we can actually seethat if we actually go in and we say OCget pod set this is pretty cooljust like this OC get pod set- oh yeah Mille you see that how the pod names got put in there okay pretty cool you look at the code you'llsee the relationship there

again we gothrough more but we have to move on tothe anvil operators right let's go ahead and do that yeah Michael do you want totalk about that yeah let's yeah

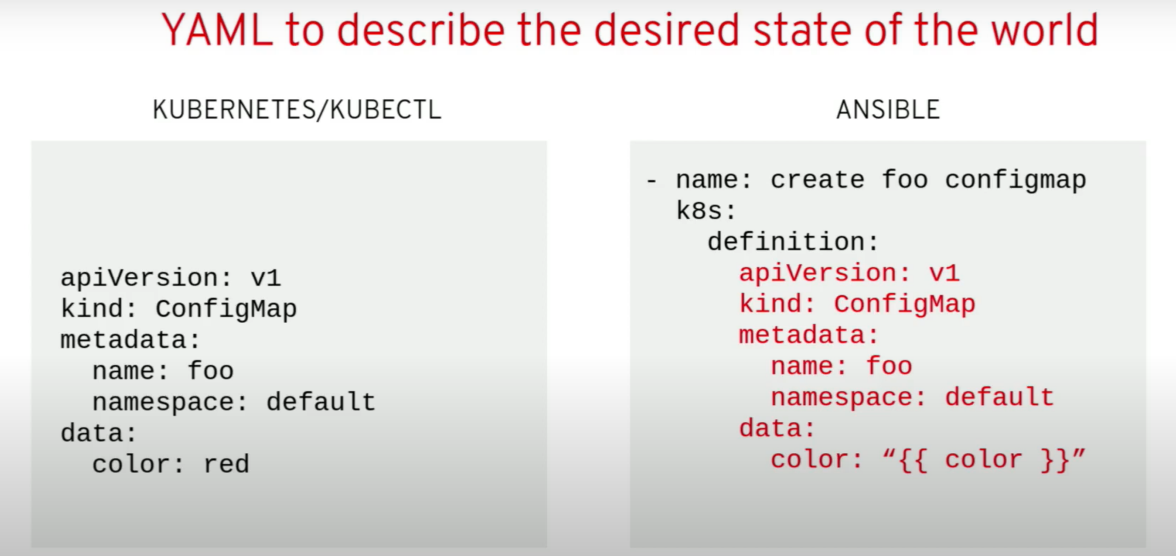
realquick guards the gentleman asked about garbage collection where is it how doesit work where can I see ityou can see it in the code but did youwant to see it in person you just say OCget pods let's look at a pod and see ifit has a garbage collection ownerreference on it I'm gonna grab it grabthe first one here I'm gonna say - ohyeah mo let's grep for the word ownerref and let's just go down see it it'sin the metadata section there it is sothese pods have owner references on themright and what does that mean it meansthat if I actually go and say OC deletepod set example pod set guess what permeating

okayall right how's this looking fantasticwe're plugged in we're ready to rocklet's look at the easy way to do thiswith ansiblenow I'm gonna apologize up front I'vegot a bit of a cough I'm gonna try mybest and not coffin to this microphone but we'll see how it goes

so why would you do this with ansible **so the Ansible both side of the operatorSDK makes itparticularly easy orders of magnitudeeasier to deploy manage applicationswith operators as opposed to doing one from scratch with even with the go** side of the operator SDK

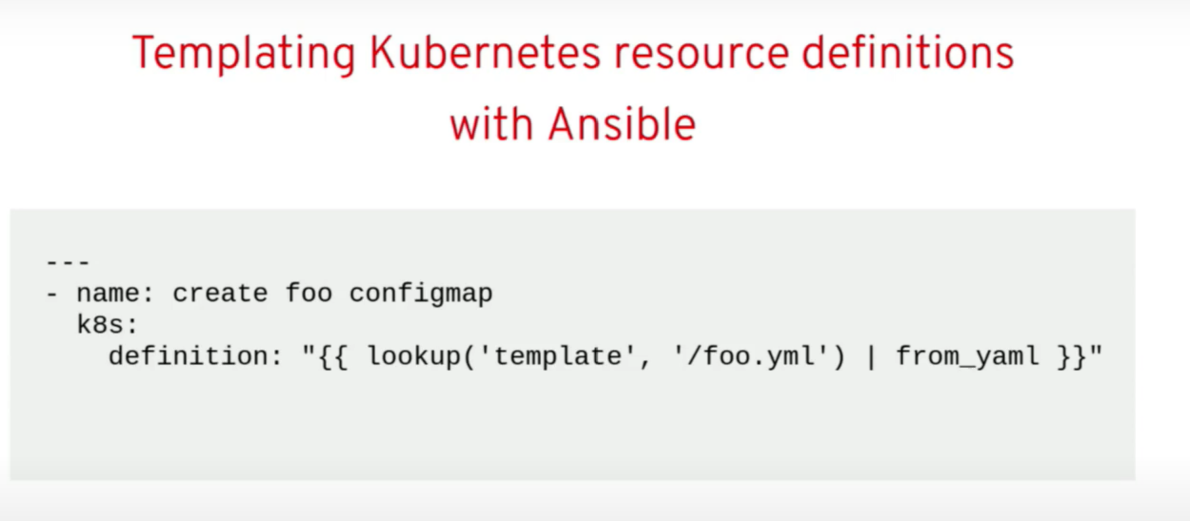
and let's look athow it fits naturally so a Kubernetes and ansible very much playing in the same kind of mindset there are both all about automation and orchestration open-source very popular active communities on both sides but two things that I like tofocus on are they're both yamo base fromtheir heavily yaml top to bottom so you're already familiar that they'reboth also all about declaring yourdesired state so in the antipas side ifyou're not familiar with ansible youcreate an answer will roll such that youcan run it over and over and over againand no matter what the starting state ofyour infrastructure was it will end upat your desired state very much like acontroller in kubernetes already doesanyway so these are a really natural anda really good fit for each other

solet's look at using ansible as a clientto interact with kubernetes so on theLeft we have normal yamo like you mightsee in a file you could cube CTL create- F this config map and you'd have aconfig map on the right side this isactual ansible code and you can see I'vejust inlinedI've directly inline to the exact sameyamo and this will using the k8s module directly create that resource or it'll ensure that it exists in the desired State so it's a little bit like like the apply verb that you're probably used to now



the one difference here you may seeis I've taken the liberty of just showing off a little bit in templateizing that last line so we'vetemplatized colorand when your ansible code runs this way inside your ansible operator any any key value pairs you put in your spec on yourCR D get plumbed in here at render timefor ansible to use so if you have color: read in your spec of your CR that youjust created in your cluster then whenansible runs its gonna use that and itsgonna render this template using thosevalues and it will ensure that thisresource has the value red as the stringred there as the color so pretty neat

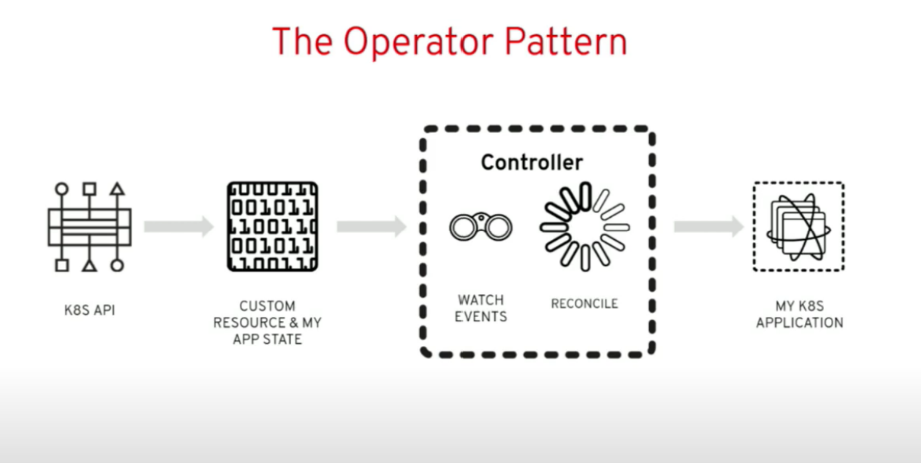
now what if you don't want to inlineyour yaml I don't really want toinline my mo either there's a better way you can maintain your yaml manifests as separate templates and templatized themjust the same way but keep them in a separate directory inside your role andwith these three lines we're lookingwe're using the lookup feature ofansible which is a stable core feature of ansible it's used in many differen tways



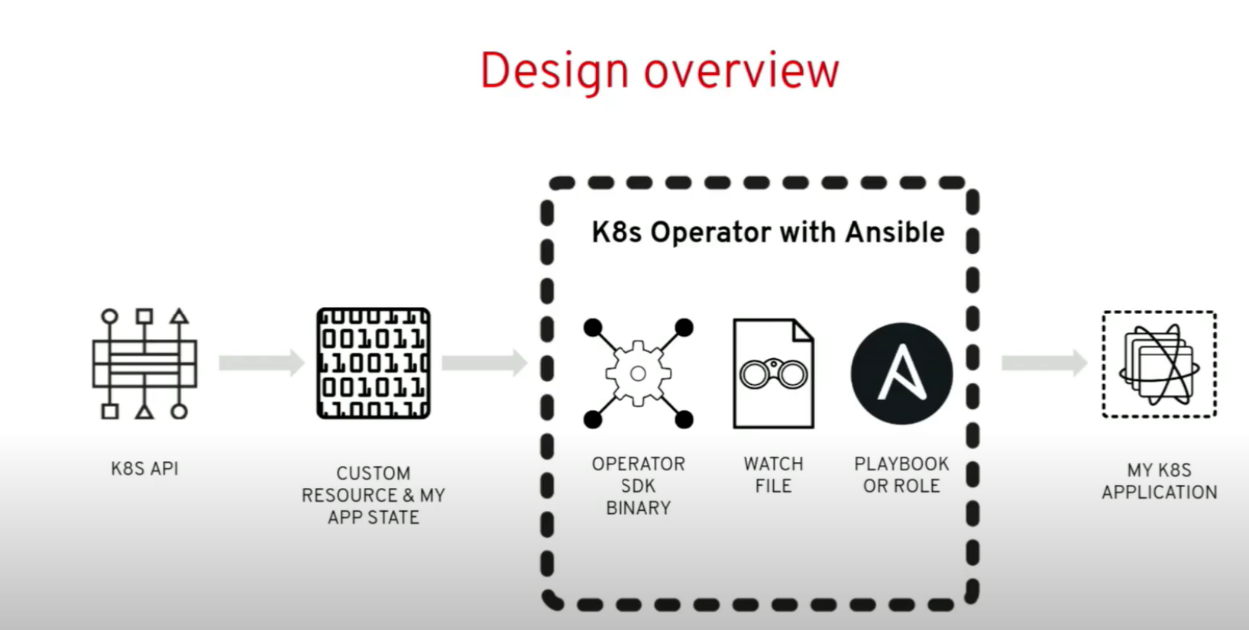
we're looking up the template thathas a certain name and we're justrendering that and the k8s module understands how to receive that as its input and now it's just going tobasically apply whatever has been rendered out of out of those templatesso this is a very natural way to interact with kubernetes from inside of ansible

so why build operators with Ansible? cutting right to the chase of these things ultimately it's it's about the ease and the ease of getting started the ease of maintaining you can have an operator with no previous experience atall even if no experience with ansible even itself up and running in under aday when your boss comes to you and sayshey we bought openshift congratulations and now you have to go make operators for these twelve things and i need them by the end of the week so we can continue to deploy them you can get thatdone actually in a big hurry and you cantake even whatever yan will manifestyou've already beenusing to deploy your workloads meet withother tooling maybe not and feed them inhere and create an operator very quickly but you still have the full power of ansible so you can do anythingfully advanced day 2 management of yourworkloads is still totally within realm

so I'll say about that for now we lookedat this already earlier so you'refamiliar now how does a controller work



Matt already talked through this we'regonna compare this is a normalcontroller and we're gonna compare thisin a moment with an answerable basedcontroller so in this normal one we'vegot this controller in the middle it'swatching events right any time a customresource or any a resource of any kindchanges that it is watching it gets thatevent and it runs that reconcilefunction right now what happens when wedrop ansible into this here's what we have



**inside that controller we have created for you a generic binary it it's written and go** using the operators 2kand it runs and it'll watch whatever resources you tell it to watch so wehave this watch file that we'll look atin a moment that really just says **when you see an event for that group, version and kind run this ansible playbook or run this ansible role** you do either one

so itreally ends up being very simple there's some other advanced features around this but then that ansible can run and doanything it wants now Matt was talkingabout owner references earlier now here this automatically puts owne rreferences on every resource you createfor you so it knows what your primaryresource is and when you createsecondary resources it will do twothings it'll put the owner reference onit and it will also start watching that other resource type also and it'll use that that handle or if you're familiar with the go side it will include of happens very naturally for you here

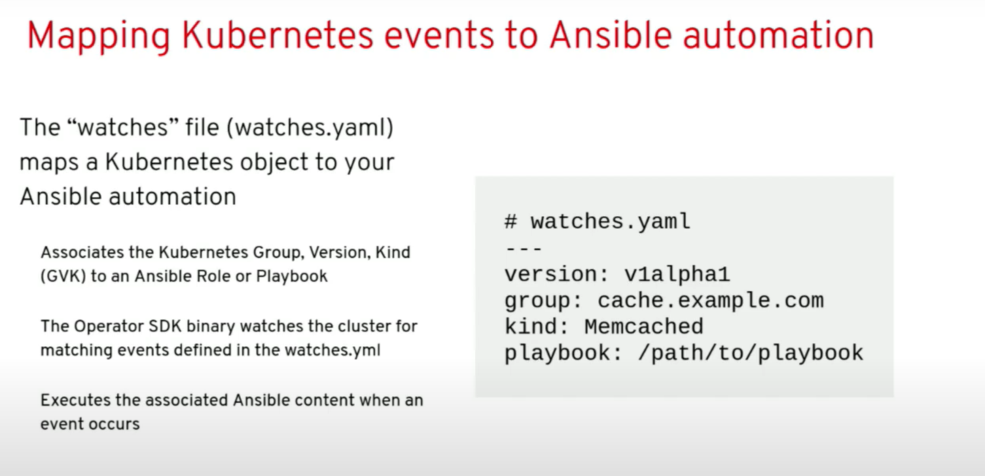
finalizar czar super easy and so on

so automating with ansibledoing this for the first time what areyou gonna do you are going to create arole or a playbook that understands howto deploy an application so if we weregonna deploy Maria DB we would create aMaria DB role and fill it out so that itunderstands how to deploy Maria DB onkubernetes

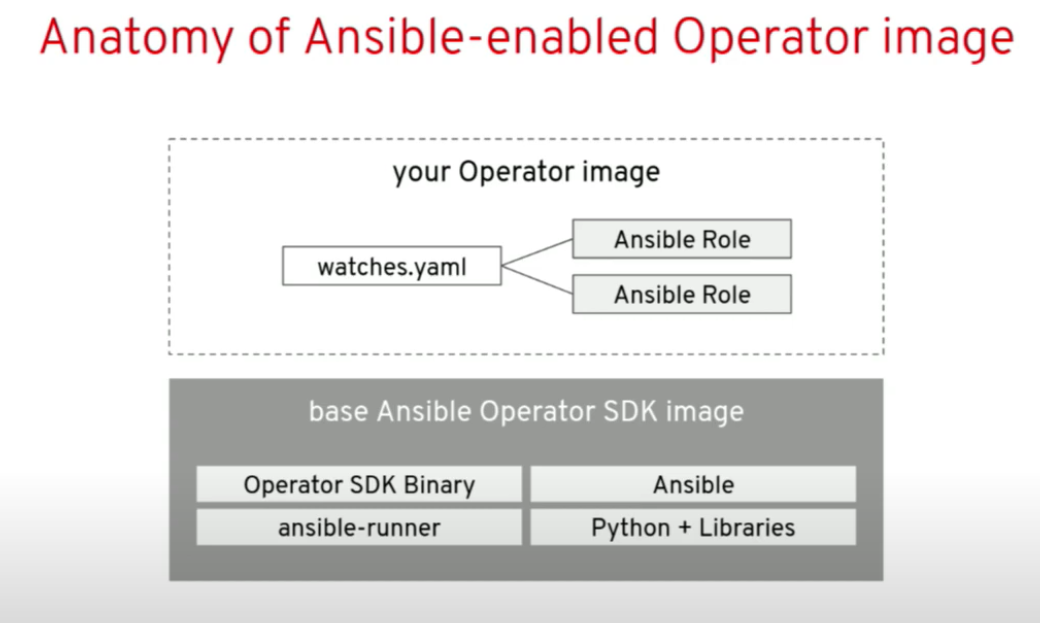
in this context in theoperator context the custom resourceinput is going to be passed into ansible at runtime so when we run answerable all that input gets passed in and is used by ansible to render templates for a logic branching you can do effectively like if kind of statements in an table you can run a block of code only when ane xpression is true so all those things you can use that input to in very powerful ways make decisions in you ransible code and

Typically you're gonna use the **k8s module** there's a collection of other related Kubernetes modules around that to do things likemanage the status and other things andthen if your interact with any external services you can use that as well we've seen somebody creating an operator with ansible to manage a physical switch that's on the top of their their rack of hosts that turn into nodes in their cluster so they're managing their kubernetes infrastructure from kubernetes including the switch and they're using Angelo's very rich collection of network equipment roles toto do that

so let's uh let's keep movingthis is what that watches file lookslike like I said really no nothing fancyhere just group version kind mapping toa playbook



and this gets scaffolded out for you by the operator K8s as you'll seenow with your custom resource in thespec so let's say we're making thisMaria DB operator we've got a Maria DB custom resource definition when I create one of those resources in the spec I canput any key value pairs I want and those get passed into ansible now what aboutthe status you have two options the easy option iswe take care of it for youthe answerable based operator willcreate a default status it's fairly generic it's not going to be specific to your workload but it is a useful and informative status it says things about what happened when ansible ran didn't encounter any errors did it change anything or not those sorts of things and then probably the last thing we lookat before we dive into the exercise thisis what the image really looks like foryour ansible based operator all thestuff on the bottom is provided for youthat binary answerable itself is in theimage all the libraries and dependenciesyou need all you add on top are thatwatches file and whatever roles areplaybooks get referenced by that watchesfile it's really very simple in terms ofartifacts



it's very little for you to manage so you can already see this is a very different picture than managing an entire go based software project yourself I'm a software engineer I love writing and go I would be happy to writegood all day long but not every organization wants to throw a full development team onto every operator that they need to make this is a very different way to manage the artifactsthat you need organizations generallyare very used to having small teams opsteams teams are not even softwareengineering oriented owning andmaintaining ansible assets as a regularmatter of course and that's just as easyto do here with your operator

okay let'sgo back to our exercises and let's seemine is alive and well now I've got goodnews for you about this exercise it's a nine step exercise the first four arestart is just reading and I already told you all that stuff so we're just gonnafast right forward past that all the wayto step five and about halfway down itsounds like they're already tearing downthe conference over thereokay that's not an earthquake I'm inEast Coaster so I'm not used to thatfear

katakoda editor

okay operator SDK newjust as you might expect right so we'regonna run operators 2k new we're sayingtype is ansible we're giving it thegroup version and kind right up frontand it just scaffold it out an entireoperator for us let's look at what wegot with this tree command right what dowe have here

okay so we have a build directory and wehave a docker file we could build this operator right now we have a workingoperator already the bad news is itdoesn't do anything but it will totallywatch our resource it will run in the antotal role that it just created every time that resource has an event there'snothing in that role yet but we alreadyhave everything we need we could run operators to get build right now andthis thing would run deployment all thisis all the usual stuff you'd get withthe SDK where there's go or ansible oranything else so we have the CR D wehave an example CR and so on all thestuff you need to deploy your operator into a cluster

**molecule** anybody familiarwith an Sable's molecule project yeahbig fans in the antipope community ofmolecule it's a whole testing frameworkit's super useful as it turns out foroperator **testing** not even just ansibleoperator testing by the way if you'reeven if you're making operators withsomething that's not ansible like an NGOor otherwise do you think molecule itturns out is a really useful way to tostand up transient cluster very quickly throw your operator into it test it and tear the whole thing down and give youvery quick feedback cycles as you're developing or your operator so that'sreally cool stuff I wish we had moretime to go into that right now

okay rolls ansible rolls so we got amemcache D roll scaffolded it out for usalready in in all this stuff there'sreally only two things you need to worryabout inside here one is tasks/main.yaml that's where we go that's like the main function in your program that's the starting point for our ansible code so that's what we're gonna we're gonna start putting ansible code and then the templates directory that's the directory we can put template files so any momanifest you already have you can juststuff them in there and reference themfrom that main dot yeah mo like we saw afew minutes ago on the slides and thenlast but not least we have this watchesdiamo file as promised that's the file where we we look and express for the group version kind run this role becausethe operator SDK already knows what groupversion kind you're gonna use it alreadycreated the role for you it has alreadyfilled out the values in that file so wedon't even need to touch it it's alreadyready to go

so let's press on so ournext step is to customize the operatorlogic cuz right now there is no logicnow rather than sit here and type outsome ansible code we're gonna take theeasy path we're gonna take the lazy path thankfully one of my colleagues DylanMurray created a role for us it's to gonna deploy memcache D so what we'regonna do here is use the ansible galaxy toto download and install that module forus an answerable galaxy that I'm thec ommand there's really nothing fancygoing on here it's just retrieving thatoff of github for us and putting that role here in our roles directory so ifwe type one-handed and look okay so inthe roles directory look we've got a wayto die I missed a step didn't I I misseda CD into my memcache D operator directory don't make that mistake like Idid and I'm glad I caught it just thenso let's run this againin let's LS in roles that looks betterokay

now we have that old role thatoriginal one that got scaffolded out forus we're just gonna delete that we don'tneed that anymore and let's take acloser look at that if we look into that role it's very similar Dillons customizes a little bit but again let's see it's got that tasks filehe doesn't actually have templates he'snot using templates here he inlined allthe ammo you can look at it yourself ifyou like but we just have that main Donllamo file and that's really the onlything you need to worry about to get anawful lot of stuff done

we can go andlook at the defaults let's see this isnot sometimes this text editor lags ahhere we gomemcache D operator roles villain memorydefaults may enamel so in the defaults we can set defaults for different thingsthis case we've got a default of size 1we'll just stick with that and you cansee here this is a printout of of what Dylan did and you can see that we'reusing just Jinja style templating here and where he's referencing so meta.name that is the name of the custom resourcethat you just created so if I had amemcache de resource in a Yemma file andI keep CTO create a shelf that thingthis operator is gonna wake up its gonna see that event it's gonna run this code and it's going to use the name of that CR to be to create the name of thisdeployment and you did the same thingin the namespace so that's pretty cooland finally the size we're using thesize also in this template again as Matt was saying this is not a brilliant operator in fact this is kind of a silly operator why would you do this we'rejust reimplemented sort of what adeployment already can do but this isthe pattern it's academic and you can see how you could add logic onto this todo more advanced things like upgradelogic fixing things when they're broken any kind of custom logic that has to happen when you scale up and down somethings like Etsy D you can't just startup new that CD processes and expect that to work there's a whole workflow of introducing new nodes into a net CDcluster for example many other databases have the same kind of thing so you canimagine how you would add that extra kind of logic to your ansible role

okaywe do need to correct our watches file because that role is in a different filepath now we just need to change this thepath on disk to that role and by the way what this is that's the path on diskinside to the container image for when you build your operatorsimage you've got a docker file for your operator you're gonna put your rolesomewhere and it's got to be whereverthis thinks it is so you just gotta makethat match

alright let's press forwardit's time to deploy this thing all rightOC did we cover what OC is great OC isjust like cube CTOokay

we created our CRD great so we've extended the kubernetes api what a miracle that is by the way how manysystems do you know that let you justextend their api with your own thingsthey're really empowered and it enablesso much it's a great feature okay we arebuilding our operator we're in theextracting phase okay we have built theoperator

now we needed to just hand wavea little bit we're gonna tell you haveto choose a name for your your operator image right your container image the sdk doesn't know it can't read your mind and know is this gonna be on docker hub isit gonna be on quay is it gonna be inyour own registry is gonna be somewhereelse so you have to pick when you build when you run that build command whereyou're gonna host like what's the full reference to your image gonna be so wejust need to put that here in our deploy manifest operator that's all we're doing here and we'll change the the poll policy just tohelp with the demo

all rightnew project we know what new project does we did that in in the last exercisethat creates basically namespace for uswith some other features around it

service account

role

role binding allvery critical things for your operator

by the way notice that operator seek ascaffolded those things out for you andthey're already ready to go with withdefaults that get you started

we don'thave to muck around and there yet withwith the our back ok it's time toactually deploy the darn thing and crossour fingers and hope that our operatorstarts successfully

and it looks like itdid ok congratulations we just made anoperator thank you thank you very forward

let's actually use thedarn thing and and just see it in action

so we have a custom resource here it's the kind is memcache D like we were looking at earlier just this is theansible implementation of that

we can golook at that file this one yes it is ok

so by default we have a size of 3 here and this is our memcache D resourcelet's uh let's just change it to two why not and let's create that CR and see what happens

oknow hopefully just now we created thatCR our ansible operator is running sothat go biner that we created for you itwas already watching memcache deresources it saw the event it noticedthe group version kind of the resourcethat the event was associated with itlooked in that watches file that mappingfile and said okay I know what role torun and it ran that role and it injectedthe entire CR all the concepts of the CRas input when it ran ansiblewith that role and when in that roleand as we saw in that role it shouldhave deployed or should have created adeployment and it did

we have ourexample memcache D memcache D we have tofind a better name for that deploymentrunning there are desired two pods thereare currently two pods they're available

life is beautiful

this thing works

Ilove it when a plan comes togetherokay

we can come here we can change thesize I don't say five oc apply eapplywe can give it a moment there we arewe've got five only two are availablebut there's five pods okay now they'reall available so it's that easy we'vetaken ansible we've taken very simple ansible code we've made it Kubernetes native as they say and a big part of being kubernetes native is plugging intothe crew Bernays api so there is now acustom API in this cluster representing memcache D and plugged into that we'replumbing all the way from that API we'recalling an ansible and we've exposedthis Angela code through that

yes siroh we we could I'm I'm trying to typeone-handed so I'm not gonna but yes wecould and I would encourage you to runthis this exercising you get home andgive that a try but it absolutely wouldif any outside actor comes and changesthis deployment that's going to generatean event and that will it'll notice that event I mean that's that is literall yactually what we just did I guess on theCR but just like it's watching this erit's also watching the deployment thatwould kick off an event and fix whateversomebody had head screwed around with

okay look we've got the memcache Dresource let's just look at the statusreal quick so like I said it's a genericstatus but it shows that the last timethis ran nothing changed two things wereokay these are ansible tasks and by theway ansible code typically is a seriesof tasks in traditional ansible it'soften you know a role to like deploy aservice on like a physical blending soyou might start by you know ensure these rpms or Debian packages are present ensure that these ports are open on the firewall ensure that some config fileexists in the right place create a system D unit file enable that system tounify all those kind of things each oneof those is a task going down your gammafile and here we're doing the exact samewith an tobacco we had two of thosetasks apparently in here and they bothran neither them changed anythingneither them failed so overall prettyboring run there of Ansible

what if all right I'm gonna put themic down for 10 seconds be right backis it on there are oh great okay thankyou Thanksokay as you get deployments we're back Ithink it already fixed it so I scaled itdone too we can look at the pods andprobably see this summer are much neweryeah see these are only three of themare only 15 seconds old so there you goscale down and the thing went right backup and did it yeah so if your interngets in there and starts messing thestuff they shouldn't mess with yourAngela based operator has your back andjust like that we can also delete thisthing get the pods so we deleted the CRand it's uh did I loops I deleted theability the operator instead oh well Ithink we've made the point the next partof this exercise we're not going to gothrough right now but you can go throughon your own is a way to run youroperator locally when you're developingand iterating on your operator whetherit's go or anta below it slows you downif you've got to build a new image andpush the image of registry and then waitfor your cluster to pull that image anddeploy that in a in a pod so theoperators ticket is a run local commitor up local command that you can use torun the operator like locally on yourlaptop and you can get very very quickfeedback cycles that way and so the restof this exercise just goes through that now when you get home this weekend if you're just so excited about the this whole week you just gotta make yourself an operator you can dothat just go to learn openshift commw ithout the training part on the URL andall these exercises are there it's not there's not enough capacity there for400 of your best friends to do it with you so that's not why we didn't use it today but you can individually do thatanytime you want there's no signup there's no information collected it'sjust totally free and open anybody cando it

thank you yeplots of other operator exercises onthere so go learn we're gonna stickaround I think they might kick us out ofhere so we might just move right out there but come ask us all your questionsw e're here you thank you for coming[Applause]Up nextAUTOPLAY