

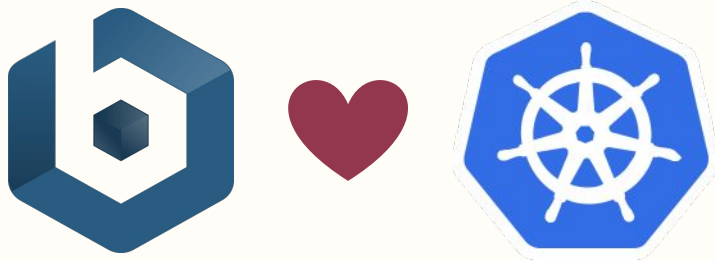
# TELEPRESENCE

SEAMLESS DEVELOPMENT ENVIRONMENTS IN KUBERNETES



bitnami

Adnan Abdulhusein (@prydonius)  
Rafael Schloming (@rschloming)



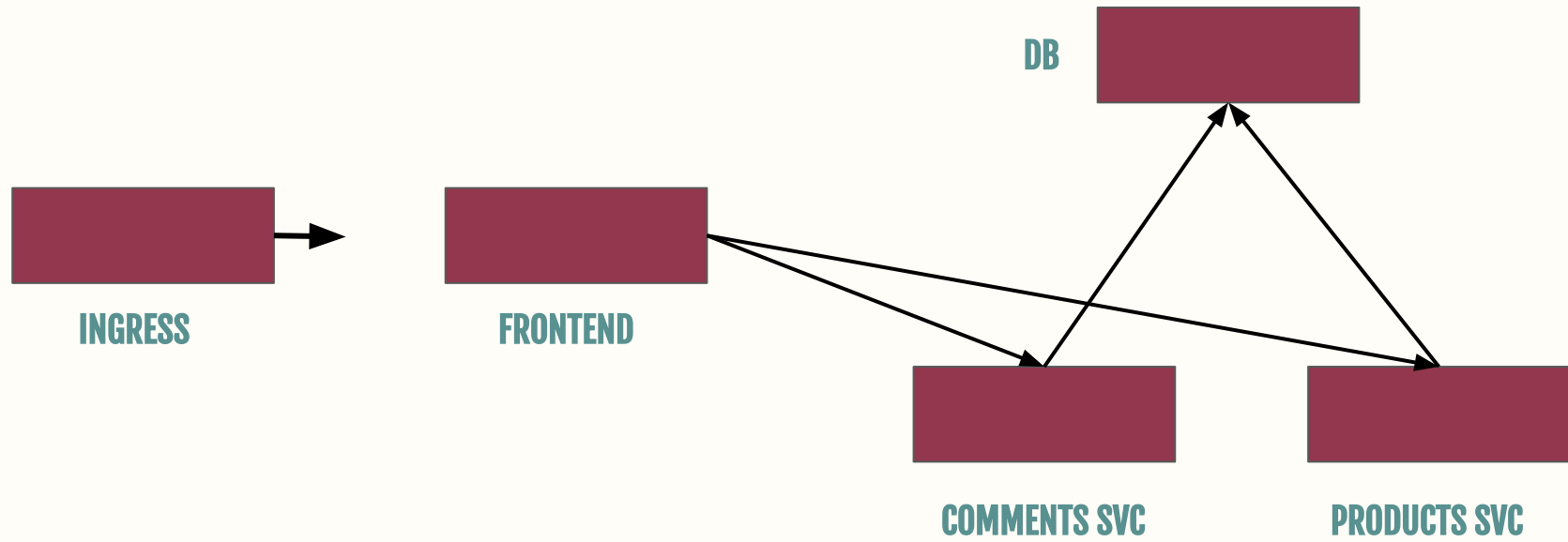
- **LEADER IN APPLICATION PACKAGING**
- **KUBERNETES CERTIFIED SERVICE PROVIDER**
- **CO-DEVELOP WITH MSFT, DEIS, HEPTIO**
- **MAINTAINING MANY OFFICIAL HELM CHARTS**



**KUBERNETES IS A GREAT  
PRODUCTION ENVIRONMENT,**

**but what about dev?**

# EXAMPLE APPLICATION



# DEVELOPING WITH DOCKER COMPOSE

```
services:
  ui:
    tty: true
    build: ./docker
    ports:
      - 4200:4200
    volumes:
      - ./app
    command: "ng serve"
  api:
    tty: true
    image: bitnami/monocular-api:latest
    volumes:
      - ./config.yaml:/config/monocular.yaml
    environment:
      - ENVIRONMENT=development
      - MONOCULAR_AUTH_SIGNING_KEY=secret
```

```
ratesvc:
  image: kubeapps/ratesvc:latest
  environment:
    - JWT_KEY=secret
  command:
    - /ratesvc
    - --mongo-url=mongodb://root@mongodb
    - --mongo-database=ratesvc
  mongodb:
    image: bitnami/mongodb:3
    environment:
      - MONGODB_ROOT_PASSWORD=password123
  auth:
    image: kubeapps/oauth2-bitnami:latest
    ...
  volumes:
    monocular-data:
```

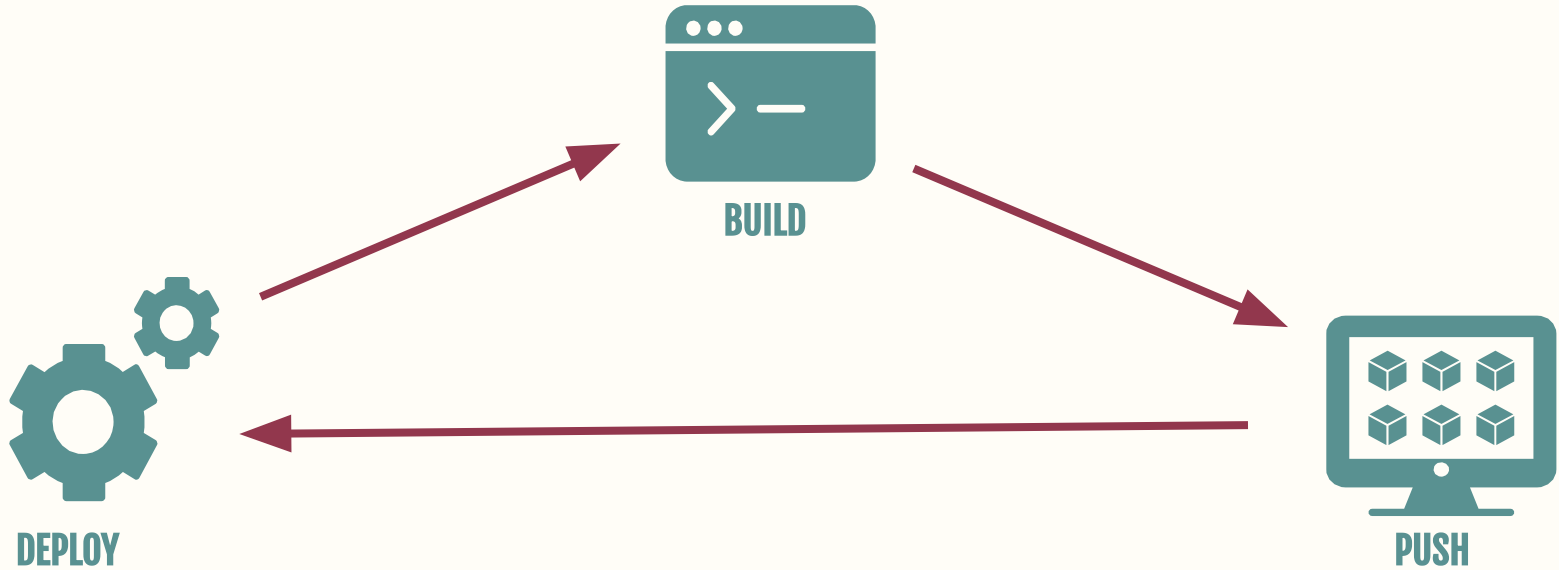




# OTHER THINGS WE NEED

- **INGRESS**
- **JOB / CRONJOB**
- **INIT CONTAINERS**
- **MOUNT KUBECONFIG  
FOR EXTENSIONS**

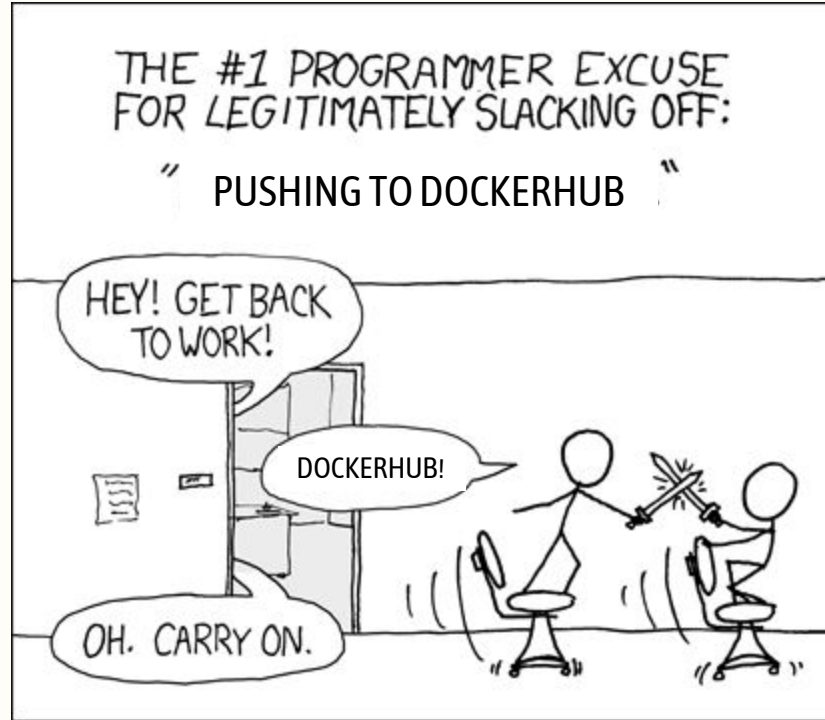
# DEVELOPMENT CYCLE

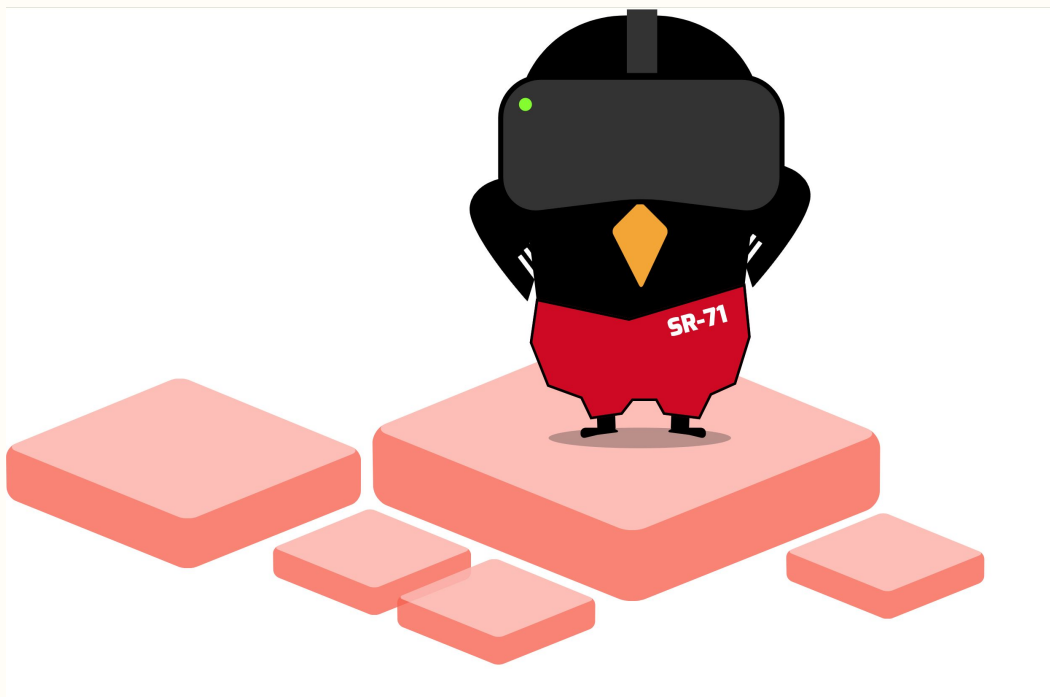




THE #1 PROGRAMMER EXCUSE  
FOR LEGITIMATELY SLACKING OFF:

" PUSHING TO DOCKERHUB "





# TELEPRESENCE

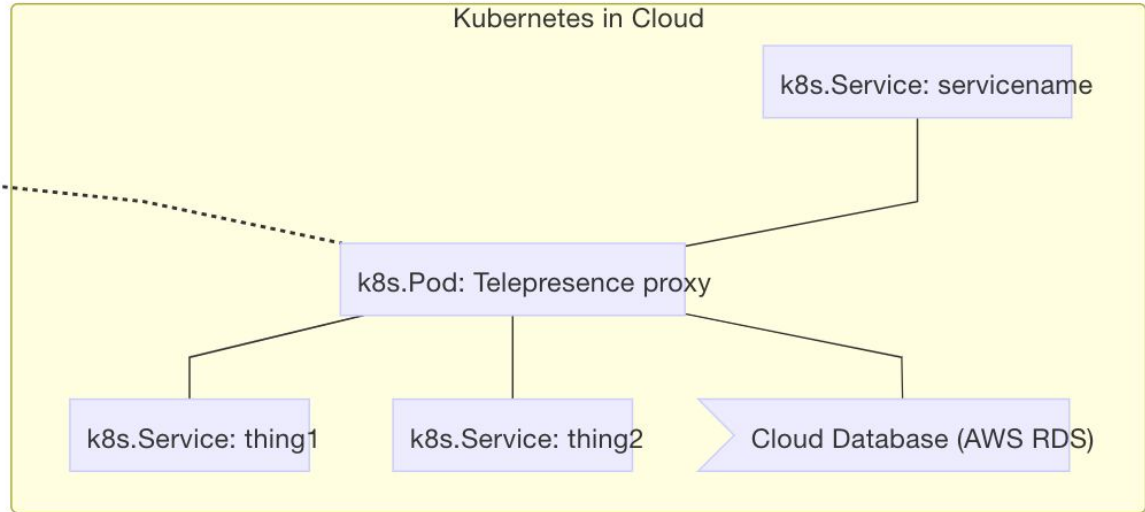
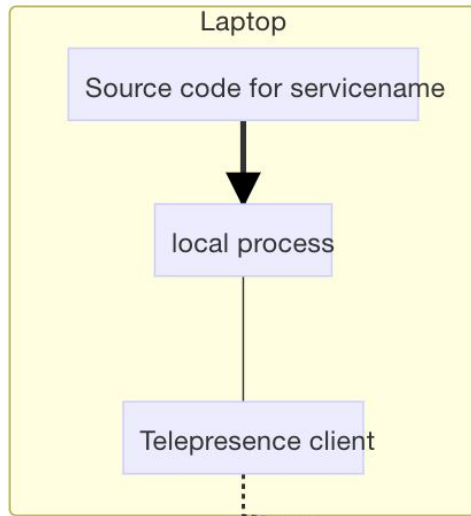
(telepresence.io)

BY



DATAWIRE

# HOW IT WORKS



**YOU GET SERVICE DISCOVERY, VOLUMES  
AND THE SAME ENVIRONMENT...**

**... FROM YOUR LOCAL MACHINE**

# PROXYING METHODS

## VPN

- ❑ WORKS BEST WITH GO
- ❑ DOESN'T WORK WITH OTHER VPNS

## INJECT-TCP

- ❑ INJECTS A SHARED LIBRARY INTO THE PROCESS
- ❑ DOESN'T WORK WITH STATICALLY LINKED

## DOCKER

- ❑ ALLOWS TO DO DOCKER-RUN
- ❑ USES DOCKER NETWORKING
- ❑ IDEAL FOR A CONTAINER-NATIVE DEVELOPMENT

# DEMO

```
    2170 function(scope, element, attr, ngSwitchController) {
    2171     var selected = attr.ngSwitch || attr.on,
    2172         selectedTranscludes = [],
    2173         selectedElements = [],
    2174         previousElements = [],
    2175         selectedScopes = [];
    2176
    2177     scope.$watchWatchExpr, function ngSwitchWatchAction(value) {
    2178         var i, ii;
    2179         for (ii = 0, ii = previousElements.length; i < ii; ++i) {
    2180             previousElements[i].remove();
    2181         }
    2182         previousElements.length = 0;
    2183
    2184         for (ii = 0, ii = selectedScopes.length; i < ii; ++i) {
    2185             var selected = selectedElements[i];
    2186             selectedScopes[i].destroy();
    2187             previousElements[i] = selected;
    2188             animate.leave(selected, function() {
    2189                 previousElements.splice(i, 1);
    2190             });
    2191         }
    2192     };
    2193
    2194     selectedElements.length = 0;
    2195     selectedScopes.length = 0;
```

# ROADMAP

## ROBUSTNESS & SPEED

- ❑ RECONNECT
- ❑ BETTER CLEANUP ON EXIT
- ❑ STARTUP SPEED
- ❑ RBAC
- ❑ REMOVE INJECT-TCP

## FUTURE

- ❑ TRAFFIC SHADOWING
- ❑ SIMPLER UX
- ❑ IDE INTEGRATIONS
- ❑ ISTIO SUPPORT



```
1140 function scope, element, attr, ngSwitchController) {  
1141   scope.$watch(attr.ngSwitch || attr.on,  
1142     function(scope, element, attr, ngSwitchController) {  
1143       selectedTranscludes = [],  
1144       selectedElements = [],  
1145       previousElements = [],  
1146       selectedScopes = [];  
1147     }, true);  
1148     function select(value) {  
1149       var selected = ngSwitchSelect(value, scope, element, attr, ngSwitchController);  
1150       if (selected !== selected) {  
1151         // We're switching, so we need to clean up the previous selection  
1152         for (ii = 0, ii = selectedScopes.length; ii < ii; ++ii) {  
1153           var selected = selectedElements[ii];  
1154           selectedScopes[ii].$destroy();  
1155           previousElements[ii] = selected;  
1156           animate.leave(selected, function() {  
1157             previousElements.splice(ii, 1);  
1158           });  
1159         }  
1160         selectedElements.length = 0;  
1161         selectedScopes.length = 0;  
1162       }  
1163     }  
1164   }  
1165 }
```

# THANK YOU



bitnami

telepresence.io  
kubernetes.com