

Replatforming to Kubernetes  
with  
**Move2Kube**



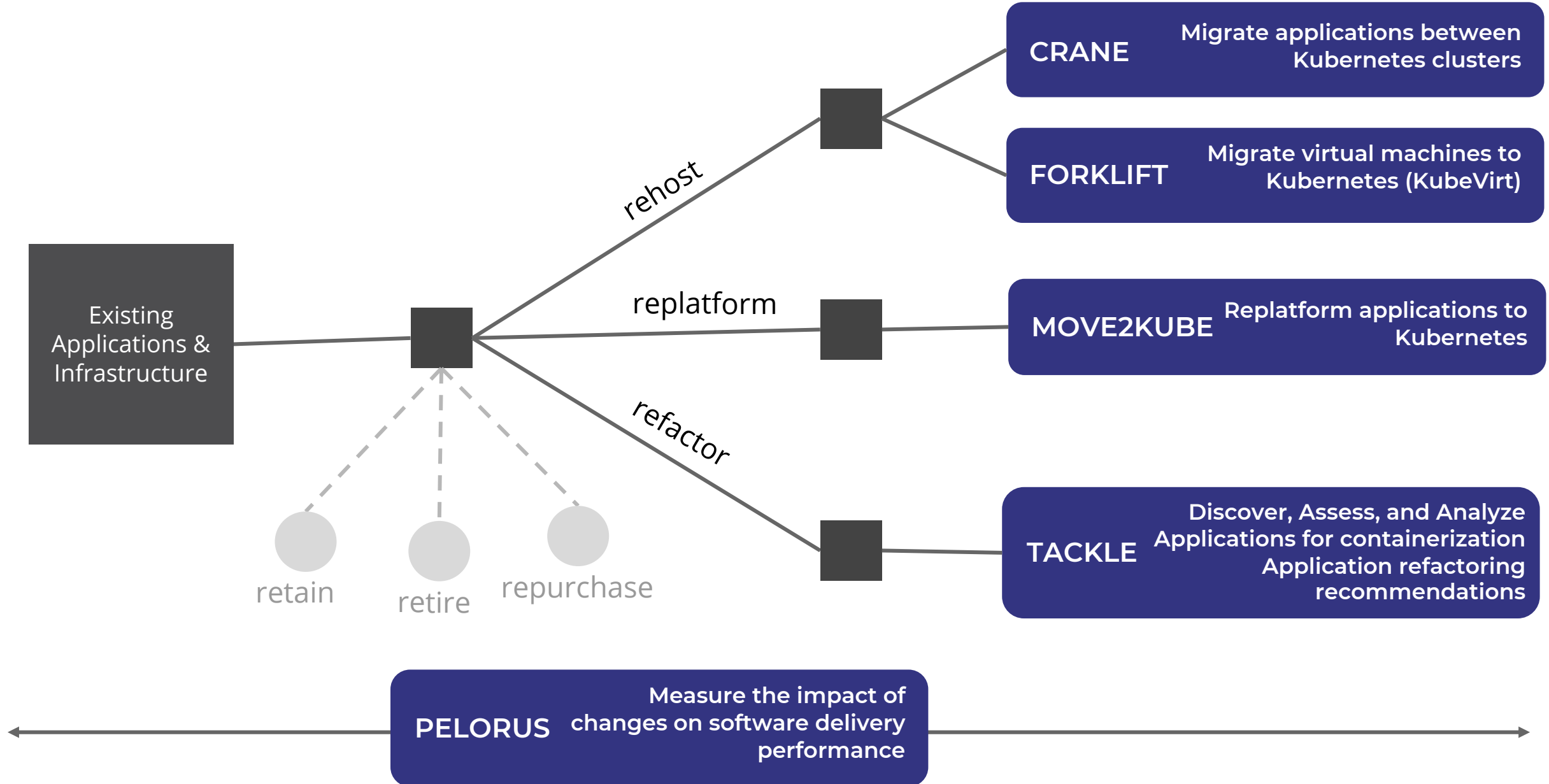
# Accelerate your journey to Kubernetes with the Konveyor Community

A community of people passionate about helping others modernize and migrate their applications to the hybrid cloud by **building tools and best practices on how to accelerate their journey to Kubernetes.**



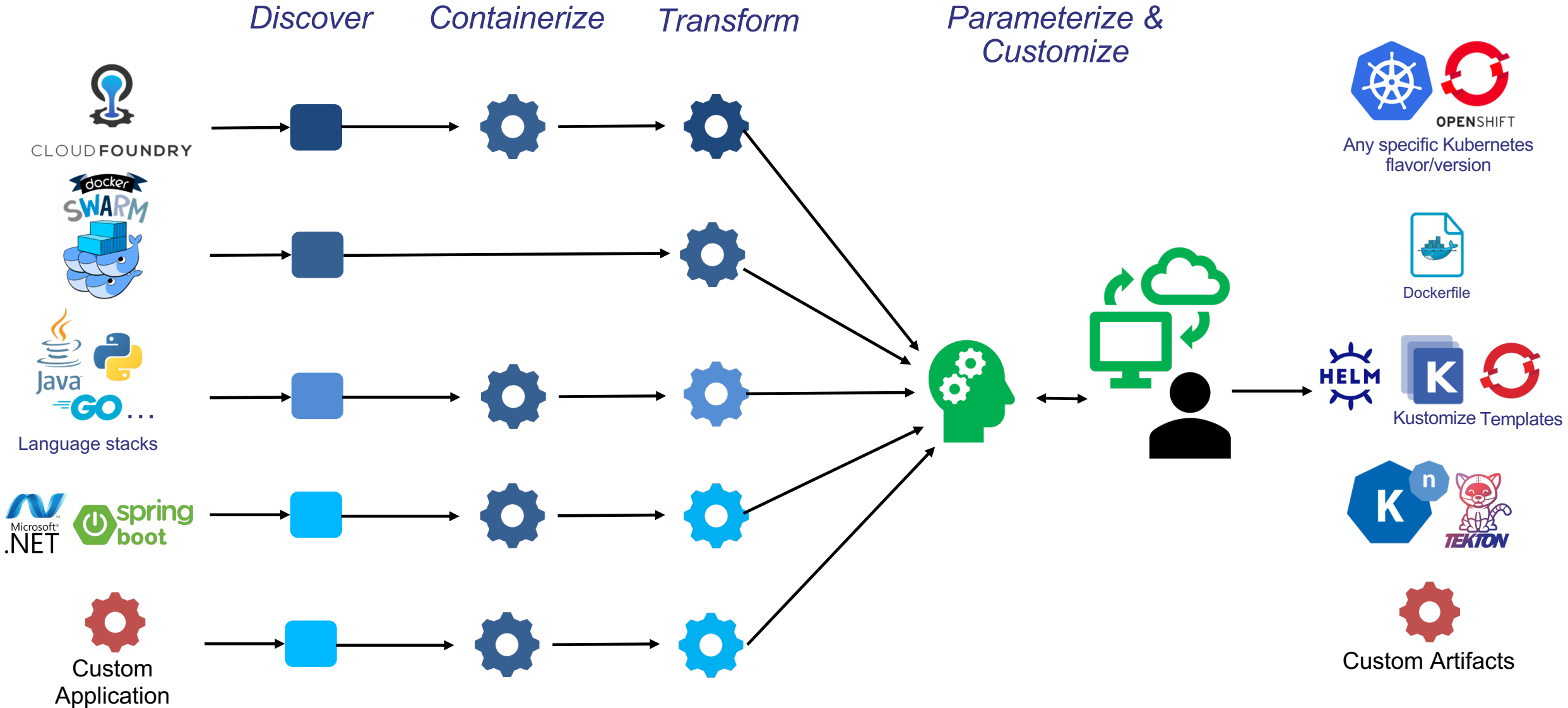
[www.konveyor.io](http://www.konveyor.io)

# Konveyor Community Projects



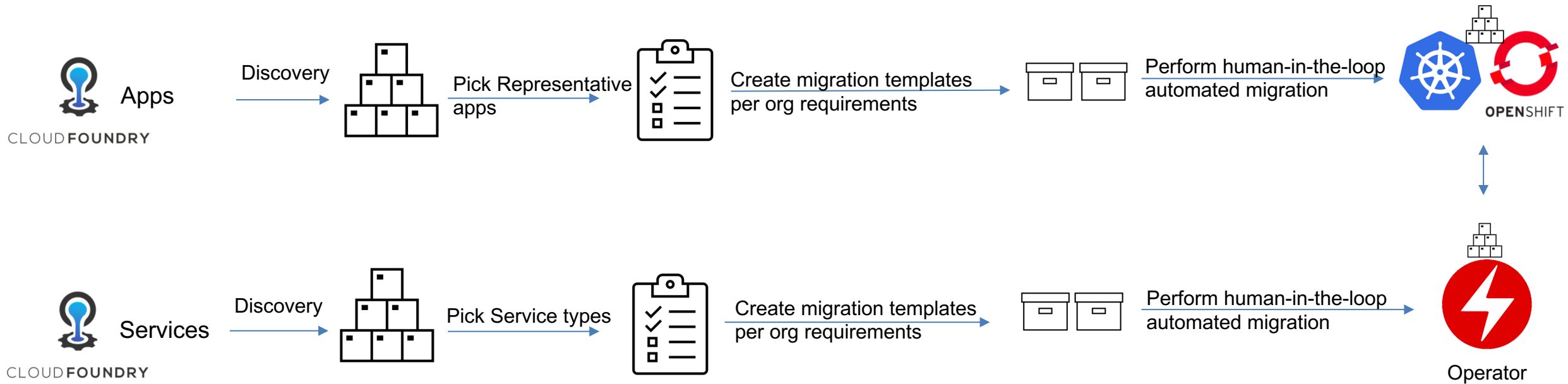
# Konveyor Move2Kube

<https://move2kube.konveyor.io/>



*Move2Kube allows you to create all your Infrastructure as Code artifacts as per your organizational requirements. It allows **integrated discovery, containerization, transformation, parameterization and customization.***

# Konveyor Move2Kube Aided Migration



# Usage modes

## Command line tool

```
ashok@move2kube-demos ashok$ move2kube plan -s samples/unified-flow/
INFO[0000] Planning Translation
INFO[0000] [*source.DockerfileTranslator] Planning translation
INFO[0000] [*source.DockerfileTranslator] Done
INFO[0000] [*source.ComposeTranslator] Planning translation
INFO[0000] [*source.ComposeTranslator] Done
INFO[0000] [*source.CfManifestTranslator] Planning translation
INFO[0006] [*source.CfManifestTranslator] Done
INFO[0006] [*source.KnativeTranslator] Planning translation
INFO[0006] [*source.KnativeTranslator] Done
INFO[0006] [*source.KubeTranslator] Planning translation
INFO[0006] [*source.KubeTranslator] Done
INFO[0006] [*source.Any2KubeTranslator] Planning translation
INFO[0020] [*source.Any2KubeTranslator] Done
INFO[0020] Translation planning done
INFO[0020] Planning Metadata
INFO[0020] [*metadata.ClusterMDLoader] Planning metadata
INFO[0020] [*metadata.ClusterMDLoader] Done
INFO[0020] [*metadata.K8sFilesLoader] Planning metadata
INFO[0020] [*metadata.K8sFilesLoader] Done
INFO[0020] [*metadata.QACacheLoader] Planning metadata
INFO[0020] [*metadata.QACacheLoader] Done
INFO[0020] Metadata planning done
INFO[0020] Plan can be found at [m2k.plan].
```

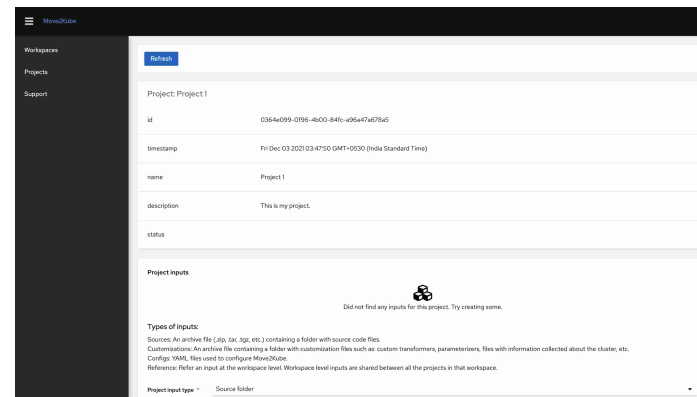


bash <(curl https://raw.githubusercontent.com/konveyor/move2kube/main/scripts/install.sh)



brew tap konveyor/move2kube  
brew install move2kube

## Web Interface



Operator



<https://github.com/konveyor/move2kube>

# One step usage

```
$ move2kube transform -s src/
```



Target-artifacts



Edit CI/CD Pipeline



Deploy

Next Steps...

# Involved Usage

**Plan** generates a plan file containing a transformation proposal (including containerization options) for all services discovered from various sources.

**Inputs:** src – Directory containing source code and collected artifact files.

**Outputs:** Plan file

Analyze code & collected artifacts and correlate

Step 1:

```
$ move2kube plan -s src/
```

Optional:

```
$ move2kube collect
```

Scrape from source and target runtime environments

**Collect** crawls metadata about the source and target runtime environments such as:

- Supported object kinds in cluster
- Apps running in cloud foundry instance
- Meta-information from local docker images.

**Inputs:** The terminal context should have cf and kubectl logged in.

**Outputs:** Data from runtime instances as files.

**Transform** transforms the input source artifacts. as per the generated plan, into target artifacts containing:

**Inputs:** Plan file and src.

**Outputs:**

- Scripts for containerization.
- Helm chart, Kustomize, Openshift templates, docker-compose.

Optimize and Translate

Step 2:

```
$ move2kube transform
```

**Move2Kube**

Target-artifacts

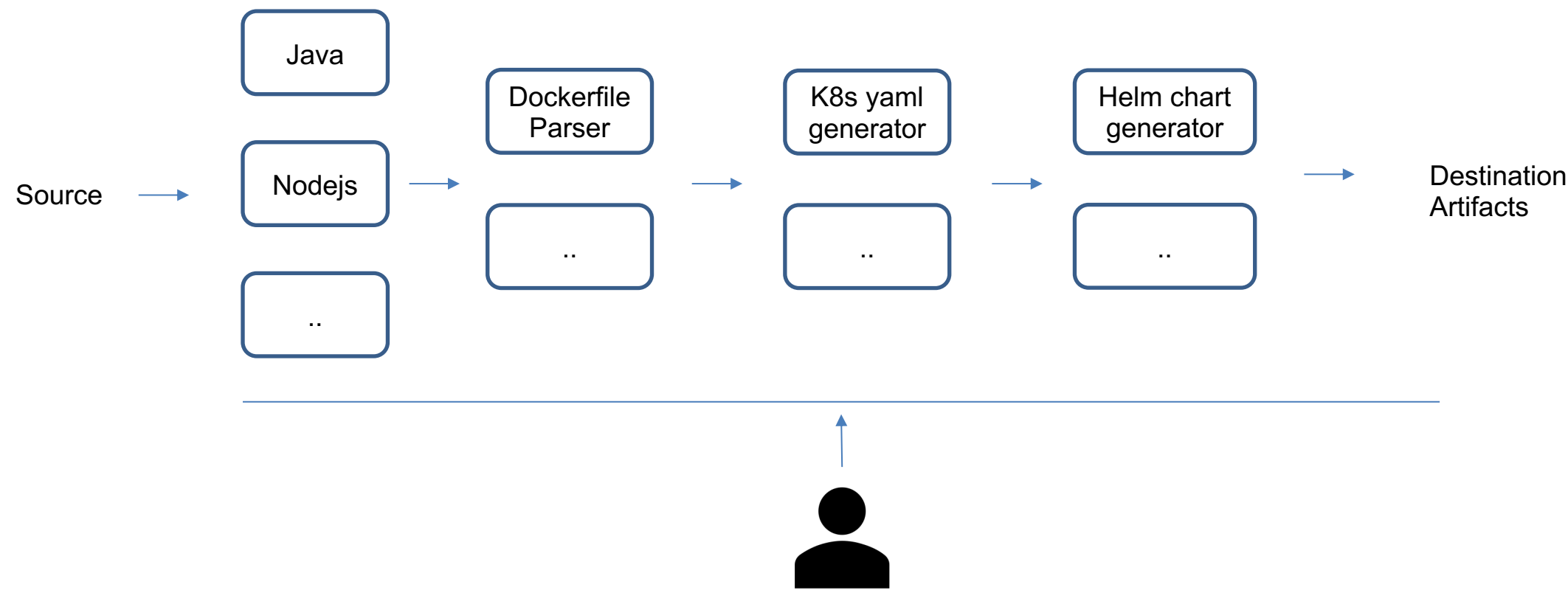
Edit CI/CD Pipeline

Deploy

Next Steps...



# Move2Kube Transformer Framework



# Customizing Transformers

```
apiVersion: move2kube.konveyor.io/v1alpha1
kind: Transformer
metadata:
  name: KubernetesForFolderChange
  labels:
    move2kube.konveyor.io/inbuilt: true
spec:
  class: "Kubernetes"
  directoryDetect:
    levels: 0
  consumes:
    IR:
      merge: true
  produces:
    KubernetesYamls:
      disabled: false
  dependency:
    matchLabels:
      move2kube.konveyor.io/kubernetesclusterselector: "true"
  config:
    outputPath: "yamls-elsewhere"
    ingressName: "{{ .ProjectName }}"
```

## Customizing in-built transformers

Each transformer exposes configurations which can be used to customize the transformer

```
18 def transform(new_artifacts, old_artifacts):
19     pathMappings = []
20     artifacts = []
21
22     for v in new_artifacts:
23         yamlsPath = v["paths"]["KubernetesYamls"][0]
24         serviceName = v["name"]
25         v["artifact"] = "KubernetesYamls"
26         artifacts.append(v)
27         fileList = fs.readdir(yamlsPath)
28         yamlsBasePath = yamlsPath.split("/")[-1]
29         # Create a path template for the service
30         pathTemplateName = serviceName.replace("-", "") + yamlsBasePath
31         tplPathData = {'pathTemplateName': pathTemplateName}
32         pathMappings.append({'type': 'PathTemplate', \
33                             'sourcePath': "({ OutputRel \"" + yamlsPath + "\"})", \
34                             'templateConfig': tplPathData})
35
36         for f in fileList:
37             filePath = fs.pathjoin(yamlsPath, f)
38             s = fs.read(filePath)
39             yamlData = yaml.loads(s)
40             if yamlData['kind'] != 'Ingress':
41                 continue
42             if 'annotations' not in yamlData['metadata']:
43                 yamlData['metadata']['annotations'] = {'kubernetes.io/ingress.class': 'haproxy'}
44             else:
45                 yamlData['metadata']['annotations']['kubernetes.io/ingress.class'] = 'haproxy'
46             s = yaml.dumps(yamlData)
47             fs.write(filePath, s)
48             pathMappings.append({'type': 'Default', \
49                                 'sourcePath': yamlsPath, \
50                                 'destinationPath': "({ \"" + pathTemplateName + "\"})"})
51
52     return {'pathMappings': pathMappings, 'artifacts': artifacts}
```

## Custom transformers in starlark

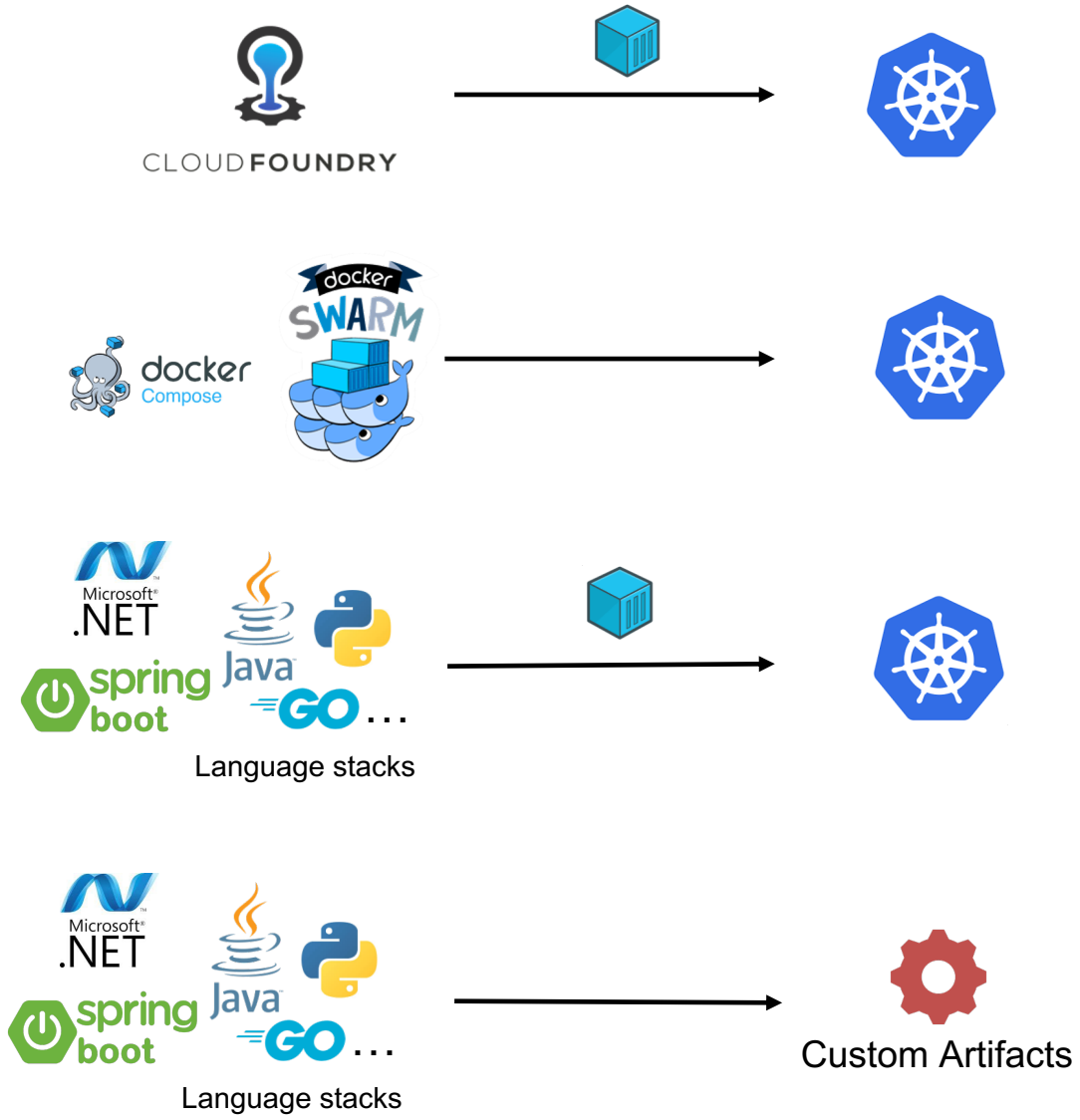
A completely functional transformer can be written in starlark (python like)

```
apiVersion: move2kube.konveyor.io/v1alpha1
kind: Transformer
metadata:
  name: ContainerizedIngressAnnotator
  labels:
    move2kube.konveyor.io/inbuilt: false
spec:
  mode: "Container"
  class: "Executable"
  consumes:
    KubernetesYamls:
      merge: false
      mode: "MandatoryPassThrough"
  produces:
    KubernetesYamls:
      disabled: false
  config:
    transformCMD: ["python", "./ingress-annotator.py"]
    container:
      image: containerized-ingress-annotator:latest
      build:
        dockerfile: "Dockerfile"
      context: "."
```

## Custom transformers as container

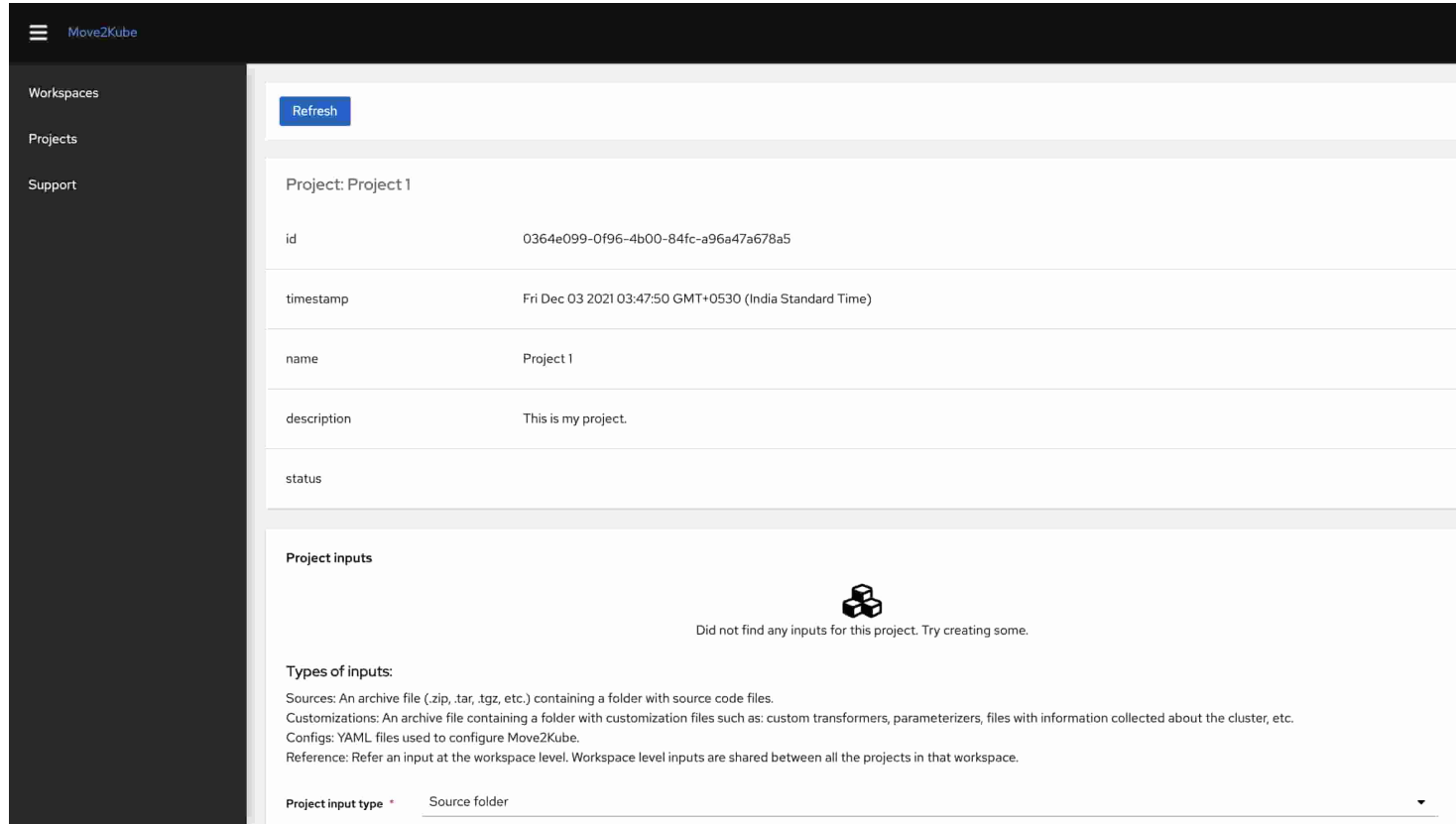
A completely functional transformer can be written in any language, and can be packaged as a container

# Sample Usecases



<https://move2kube.konveyor.io/tutorials>

# Web UI – with Authorization and multi-tenancy

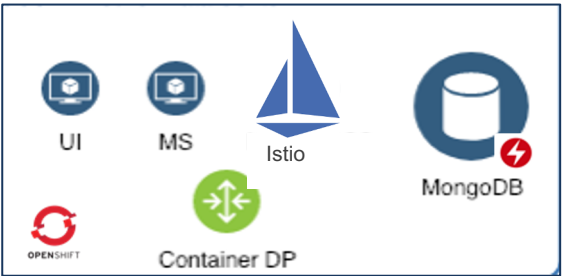


```
docker run --rm -it -p 8080:8080 -v "${PWD}/move2kube-api-data:/move2kube-api/data" quay.io/konveyor/move2kube-ui:latest
```

# Move2Kube Case Study



Application and middleware services on Docker Swarm



Application and middleware service on Red Hat Openshift

Tasks	Manual (est.)	Move2Kube
Discover relevant assets	3 days	3 hours
Direct artifact translations	14 days	1 day
Complex artifact translations	32 days	4 days
Exploit OCP features + best practices	3 days	3 hours
Customize for a deployment	4 days	30 mins
Rightsize initial config	5 days	1 day
	61 days	7 days

For ~100 container scenario.

Based on extrapolation from PoC experience on a limited subset of the migration process, we will further evolve this from actual delivery data

← ~9x →  
reduction

# Contributions are welcome!

## Head over and submit PRs



<https://github.com/konveyor/move2kube>  
<https://github.com/konveyor/move2kube-api>



React



<https://github.com/konveyor/move2kube-ui>



<https://github.com/konveyor/move2kube-operator>



<https://github.com/konveyor/move2kube-tests>



<https://github.com/konveyor/homebrew-move2kube>

github.com/konveyor/move2kube

Search or jump to... Pull requests Issues Marketplace Explore

konveyor / move2kube

Unwatch 14 Unstar 40 Fork 18

Code Issues 13 Pull requests 1 Actions Projects Wiki Security Insights Settings

master 1 branch 1 tag

Go to file Add file Code

ashokponkumar Add default support for generic cluster types ✓ abafe03 2 hours ago 72 commits

cmd/move2kube	Fix bug with empty git domain. Set RootDir after loading plan. Added ...	2 days ago
docs	Adding document guides to extend Move2Kube	3 days ago
imgs	remove diagram	27 days ago
internal	Add default support for generic cluster types	2 hours ago
misc	initial working version of move2kube	28 days ago
samples	initial working version of move2kube	28 days ago
scripts	Fixed bug with paths and hashes. Use cobra instead of os.Args	8 days ago
types	More idiomatic code. Added clean to make ci target. Build before test...	2 hours ago
.dockerignore	add codecov	25 days ago
.gitignore	add codecov	25 days ago
.golangci.yml	Add golangci config	14 days ago
.travis.yml	move CI routine to a make target	5 days ago
Dockerfile	Fix license in Dockerfile	21 days ago
LICENSE	initial working version of move2kube	28 days ago
Makefile	More idiomatic code. Added clean to make ci target. Build before test...	2 hours ago
README.md	Add prerequisites to README.md	2 hours ago
USAGE.md	Updated usage instructions	15 days ago
code-of-conduct.md	Update slack ref in usage and code of conduct	20 days ago

About

Move2Kube is a command-line tool that accelerates the process of re-platforming to Kubernetes/OpenShift.

kubernetes openshift modernization ibm-research replatform hacktoberfest

Readme

Apache-2.0 License

Releases

1 tags

Create a new release

Contributors 8

Languages

- Go 93.7%
- Shell 4.1%
- Dockerfile 1.5%
- Other 0.7%