👋 Welcome!

Deploying a Service on EDITO Datalab

Learn how to turn your script into a containerized web service and launch it on the EDITO platform.

By **Samuel Fooks**
Flanders Marine Institute (VLIZ)

🎯 What You'll Learn

- ✅ Dockerize a script (R or Python)
- ✅ Push the image to a public Docker registry
- ✅ Configure a Helm chart
- ✅ Deploy the service on the EDITO playground
- ✅ Publish to production via Merge Request

🧱 Dockerizing Your Script

Example: R Markdown → Plumber API

```
# plumber.R
#* @get /hello
function() {
  list(message = "Hello from EDITO!")
}
```

Dockerfile Example

```
FROM rstudio/plumber

COPY plumber.R /plumber.R

EXPOSE 8000

CMD ["R", "-e", "pr <- plumber::plumb('/plumber.R'); pr$run(host='0.0.0.0', port=8000)"]
```

🐳 Build and Push Docker Image

```
docker build -t ghcr.io/yourusername/hello-edito:latest .

echo $CR_PAT | docker login ghcr.io -u yourusername --password-stdin

docker push ghcr.io/yourusername/hello-edito:latest
```

Replace `yourusername` with your GitHub account.

Generate a token at https://github.com/settings/tokens.

📦 Set Up Helm Chart

```
git clone https://gitlab.mercator-ocean.fr/pub/edito-infra/service-playground.git
cd service-playground
cp -r terria-map-viewer hello-edito
cd hello-edito
```

Edit `Chart.yaml`:

```
name: hello-edito
description: A simple Plumber API demo on EDITO
home: https://github.com/yourusername/hello-edito
icon: https://your.icon.url/icon.png
keywords: [plumber, r, api]
version: 1.0.0
appVersion: "0.1.0"
```

🛠️ Update Chart Values

`values.yaml`

```
service:
  image:
    version: "ghcr.io/yourusername/hello-edito:latest"

networking:
  service:
    port: 8000
```

values.schema.json

```
"listEnum": [
  "ghcr.io/yourusername/hello-edito:latest"
],
"default": "ghcr.io/yourusername/hello-edito:latest"
```

templates/NOTES.txt

```
Your Hello EDITO API is being deployed!

It will be available on this [link](http{{ if $.Values.ingress.tls }}s{{ end }}://{{ .Values.ingress.hostname }}).
```

🌐 Enable Ingress (Optional)

In `values.schema.json`, allow user-defined ingress:

```
"x-onyxia": {
  "overwriteDefaultWith": "{{project.id}}-{{k8s.randomSubdomain}}-0.{{k8s.domain}}"
}
// Remove "hidden": true line
```

🔒 Add S3 or Marine Service Secrets (Optional)

Add to `values.schema.json`:

```
"s3": {
  "x-onyxia": { "overwriteSchemaWith": "ide/s3.json" }
}
```

Enable secret in templates:

```
envFrom:
- secretRef:
    name: {{ include "library-chart.secretNameS3" . }}
```

🚀 Launch in Playground

- Push branch to GitLab
- Wait for auto-publish (5–10 min)
- Launch from EDITO Datalab
- Test endpoint (e.g., `/hello`)

✅ Production Release

Once tested:

- Add yourself to `Chart.yaml` as maintainer
- Submit a Merge Request
- Ping `@pub/edito-infra/codeowners`

🙌 Done!

🎉 Your service is live on EDITO!

🧩 You now know how to go from script → container → Helm → Datalab.

Questions?

📧 edito-infra-dev@mercator-ocean.eu