👋 Welcome!

Deploying a Data Processing Workflow to EDITO

Learn how to turn your data processing scripts into containerized batch jobs and deploy them on the EDITO platform.

Presented by **Samuel Fooks**
Flanders Marine Institute (VLIZ)

For all the PDFs and code, check out the workshop GitHub repository

What is a Process on EDITO?

A **process** is a computational workflow that:

- Takes input data and transforms it into output data
- Performs analysis, prediction, or simulation
- Runs as a batch job (not interactive)
- Processes data through algorithms or mathematical operations

Examples:

- Machine learning models
- Statistical analysis workflows
- Data processing pipelines
- Simulation models

🎯 What We'll Go Over

- ✅ Identify when your application is a process
- ✅ Dockerize your data processing workflow
- ✅ Push the image to a container registry
- ✅ Create Helm charts for Kubernetes deployment
- ✅ Deploy to EDITO Process Playground
- ✅ Submit for production deployment

All this is also covered in EDITO Process Documentation.

Get an Account on EDITO

🌐 Become a Beta Tester:

Sign up here

🔑 Sign up to Mercator Ocean GitLab:

Create your account

Access EDITO Playgrounds

Process Playground Repository

- Process Playground Repository

Service Playground Repository

- Service Playground Repository

🐳 Step 1: Dockerize Your Process

Example Process Structure

```
my_process/
├── Dockerfile
├── requirements.txt
├── Scripts/
│   ├── 01_data_preparation.R
│   └── 02_model_analysis.R
└── README.md
```

Dockerfile Example

```
FROM rocker/r-ver:4.3.0

# Install system dependencies
RUN apt-get update && apt-get install -y \
    curl \
    libcurl4-openssl-dev \
    libssl-dev \
    && rm -rf /var/lib/apt/lists/*

# Install R packages
COPY requirements.txt /requirements.txt
RUN Rscript -e "install.packages(readLines('requirements.txt'))"

# Copy scripts
COPY Scripts/ /Scripts/

# Set working directory
WORKDIR /data

# Default command
CMD ["Rscript", "/Scripts/01_data_preparation.R"]
```

Make a container registry token

Working with container registry

You need your container registry token

🐳 Build and Push Docker Image

Build and version your container using semantic versioning docs

```
# Build the image
docker build -t ghcr.io/yourusername/my-process:1.0.0 .

# Login to registry
export CR_PAT = mycontainerregistrytoken
echo $CR_PAT | docker login ghcr.io -u yourusername --password-stdin

# Push the image
docker push ghcr.io/yourusername/my-process:1.0.0
```

Test Your Container Locally

```
# Test the container
docker run -v $(pwd)/data:/data ghcr.io/yourusername/my-process:1.0.0
```

Your working process is now usable by anyone, anywhere with Docker and an internet connection

🚀 Step 2: Deploy to EDITO Process Playground

How to add your process, README.md

Clone the Process Playground

```
git clone https://gitlab.mercator-ocean.fr/pub/edito-infra/process-playground.git
cd process-playground
git checkout -b my-process-workflow
git push origin my-process-workflow
```

Understanding Kubernetes Jobs

- **Jobs**: Run batch workloads to completion
- **Pods**: Smallest deployable units in Kubernetes, running one or more containers
- **PVCs**: Persistent Volume Claims for data storage
- **Init Containers**: Run before main containers

Process Workflow Pattern

The EDITO process template follows a simple three-stage pattern:

1. **Download**: Input data from S3 → `/data/input`
2. **Process**: Run your scripts in `/data` → output to `/data/output`
3. **Upload**: Results from `/data/output` → S3 storage

Create Your Process Directory

```
process-playground/
└── my_process_workflow/
    ├── Chart.yaml
    ├── values.yaml
    ├── values.schema.json
    └── templates/
        ├── job.yaml
        ├── pvc.yaml
        ├── secret-s3.yaml
        └── serviceaccount.yaml
```

Chart.yaml Example

```yaml
apiVersion: v2
name: my-process-workflow
description: A data processing workflow for EDITO
icon: https://example.com/icon.png
home: https://github.com/yourusername/my-process

type: application
version: 0.1.0
appVersion: "1.0.0"

dependencies:
  - name: library-chart
    version: 1.5.14
    repository: https://inseefrlab.github.io/helm-charts-interactive-services
```

values.yaml Configuration

```yaml
# Image configuration
image:
  repository: ghcr.io/yourusername/my-process
  tag: "1.0.0"
  pullPolicy: IfNotPresent

# Processing configuration
processing:
  dataPreparationCommand: "Rscript /Scripts/01_data_preparation.R"
  modelAnalysisCommand: "Rscript /Scripts/02_model_analysis.R"

# Input/Output paths
inputData:
  s3Path: "my-process/input"

output:
  s3Path: "my-process/output"
```

Key Job Template Features

- **S3 Download Init Container**: Downloads input data
- **Processing Containers**: Run your custom commands
- **S3 Upload Container**: Uploads results
- **Shared Volume**: `/data` directory for all containers
- **Resource Management**: CPU and memory limits

Simple Data Flow

The example process uses a straightforward directory structure:

- Input data is downloaded to `/data/input`
- Processing happens in `/data`
- Results are written to `/data/output`
- No complex environment variable handling needed

Deploy Your Process

1. **Add your process directory** to the playground
2. **Update the main values.yaml** to include your process
3. **Test locally** with Helm
4. **Commit and push** your changes

```
git add .
git commit -m "Added my awesome process"
# Push the changes to your branch
git push origin my-process-workflow
```

Submit a Merge Request

- Go to the Process Playground
- Create a merge request from your branch
- Wait for pipeline validation
- Once approved, your process will be available on EDITO!

Process vs Service vs Tutorial

Type	Purpose	Interaction	Example
Process	Data transformation	Batch job	ML model, data analysis
Service	Interactive application	Web interface	Dashboard, API
Tutorial	Learning content	Step-by-step	R Markdown, Jupyter

🎉 Congratulations!

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

What's Next?

- Process Playground README.md
- EDITO Datalab
- Docker Documentation
- Kubernetes Jobs Documentation

Questions?

Contact us:

- EDITO Community
- GitHub Issues

Resources:

- EDITO Documentation
- Process Playground

Thank You! 🙏

Funded by the European Union