

SonataFlow

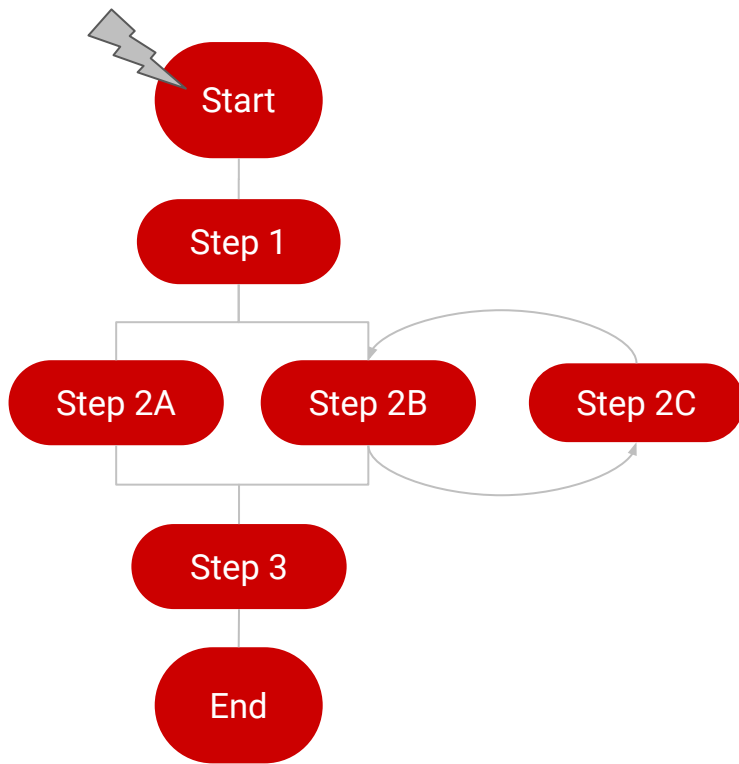
Running Serverless Workflows on Kubernetes and Knative

Gloria Ciavarrini
Senior Software Engineer



What is a workflow?

What is a workflow?



A *sequence of steps* that are executed to achieve a specific goal.

The steps in a workflow can be:

- executed in parallel or sequentially
- triggered by events
 - HTTP requests
 - messages from a queue
 - changes to a database

How to define a workflow?

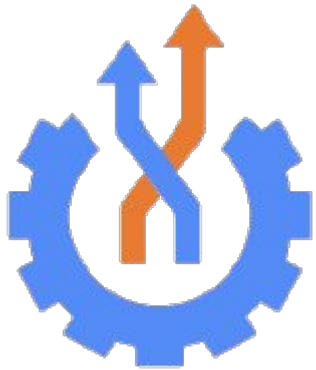
	BPMN	BPEL	WfMC	CWL	CNCF Serverless Workflow
Vendor neutrality	Yes	Yes	No	Yes	Yes
Interoperability	Yes	Yes	No	Yes	Yes
Consistency	Yes	Yes	No	Yes	Yes
Ecosystem	Mature	Mature	Mature	Growing	Growing
Best Use cases	Business processes	Business processes	Computational data analysis	Business processes	Serverless applications / Event driven applications

How to define a workflow?

	BPMN	BPEL	WfMC	CWL	CNCF Serverless Workflow
Vendor neutrality	Yes	Yes	No	Yes	Yes
Interoperability	Yes	Yes	No	Yes	Yes
Consistency	Yes	Yes	No	Yes	Yes
Ecosystem	Mature	Mature	Mature	Growing	Growing
Best Use cases	Business processes	Business processes	Computational data analysis	Business processes	Serverless applications / Event driven applications

What is CNCF Serverless Workflow Specification?

Serverless Workflow Specification



- Open-source specification for defining and running declarative workflow models (JSON/YAML)
- Sandbox Project in Cloud Native Computing Foundation (CNCF)
- Vendor neutral
- Integrations with standards for events and services applications
 - CloudEvents
 - OpenAPI
 - GraphQL
 - gRPC
 - ... and many more!

Serverless Workflow Specification



Serverless Workflow Specification

```
---  
  
id: sampleWorkflow  
version: '1.0.0'  
specVersion: '0.8'  
name: Sample Workflow  
description: Sample Workflow  
start: MyStartingState  
states: []  
functions: []  
events: []  
errors: []  
retries: []
```

Functions

Reusable declarations of functions that need to be invoked, or expressions to be evaluated.

```
---
id: sampleWorkflow
version: '1.0.0'
specVersion: '0.8'
name: Sample Workflow
description: Sample Workflow
start: MyStartingState
states: []
functions:
  - name: concatA
    operation: .result|=."A"
    type: expression
  - name: greetFunction
    type: custom
    operation: sysout
  - name: getBranch
    type: custom
    operation: rest:get:https://api.github.com/myrepo/myfile.txt
  - name: execute
    type: custom
    operation: "service:java:com.my.project.workflow.BasicClass::execute"
events: []
errors: []
retries: []
```

Events

Reusable declarations of events that need to be consumed to start or continue workflow instances, trigger function/service execution, or be produced during workflow execution.

```
---
id: sampleWorkflow
version: '1.0.0'
specVersion: '0.8'
name: Sample Workflow
description: Sample Workflow
start: MyStartingState
states: []
functions: []
events:
  - name: startEvent
    source: ""
    kind: produced
    type: start
  - name: moveEvent
    source: cloudEventGitHubSource
    kind: consumed
    type: com.github.pull_request.opened
    correlation:
      - contextAttributeName: repository
        contextAttributeValue: myrepo
errors: []
retries: []
```

Events

Reusable declarations of events that need to be consumed to start or continue workflow instances, trigger function/service execution, or be produced during workflow execution.

```
---
id: sampleWorkflow
version: '1.0.0'
specVersion: '0.8'
name: Sample Workflow
description: Sample Workflow
start: MyStartingState
states: []
functions: []
events:
  - name: startEvent
    source: ""
    kind: produced
    type: start
  - name: moveEvent
    source: cloudEventGitHubSource
    kind: consumed
    type: com.github.pull_request.opened
    correlation:
      - contextAttributeName: repository
        contextAttributeValue: myrepo
errors: []
retries: []
```

Events

Reusable declarations of events that need to be consumed to start or continue workflow instances, trigger function/service execution, or be produced during workflow execution.

Event correlation rules for this event.
Only used for consumed events

```
---
id: sampleWorkflow
version: '1.0.0'
specVersion: '0.8'
name: Sample Workflow
description: Sample Workflow
start: MyStartingState
states: []
functions: []
events:
  - name: startEvent
    source: ""
    kind: produced
    type: start
  - name: moveEvent
    source: cloudEventGitHubSource
    kind: consumed
    type: com.github.pull_request.opened
    correlation:
      - contextAttributeName: repository
        contextAttributeValue: myrepo
errors: []
retries: []
```

States

Workflow states define building blocks of the workflow execution instructions. They define the control flow logic instructions on what the workflow is supposed to do. States can reference the reusable function, event and retry definitions.

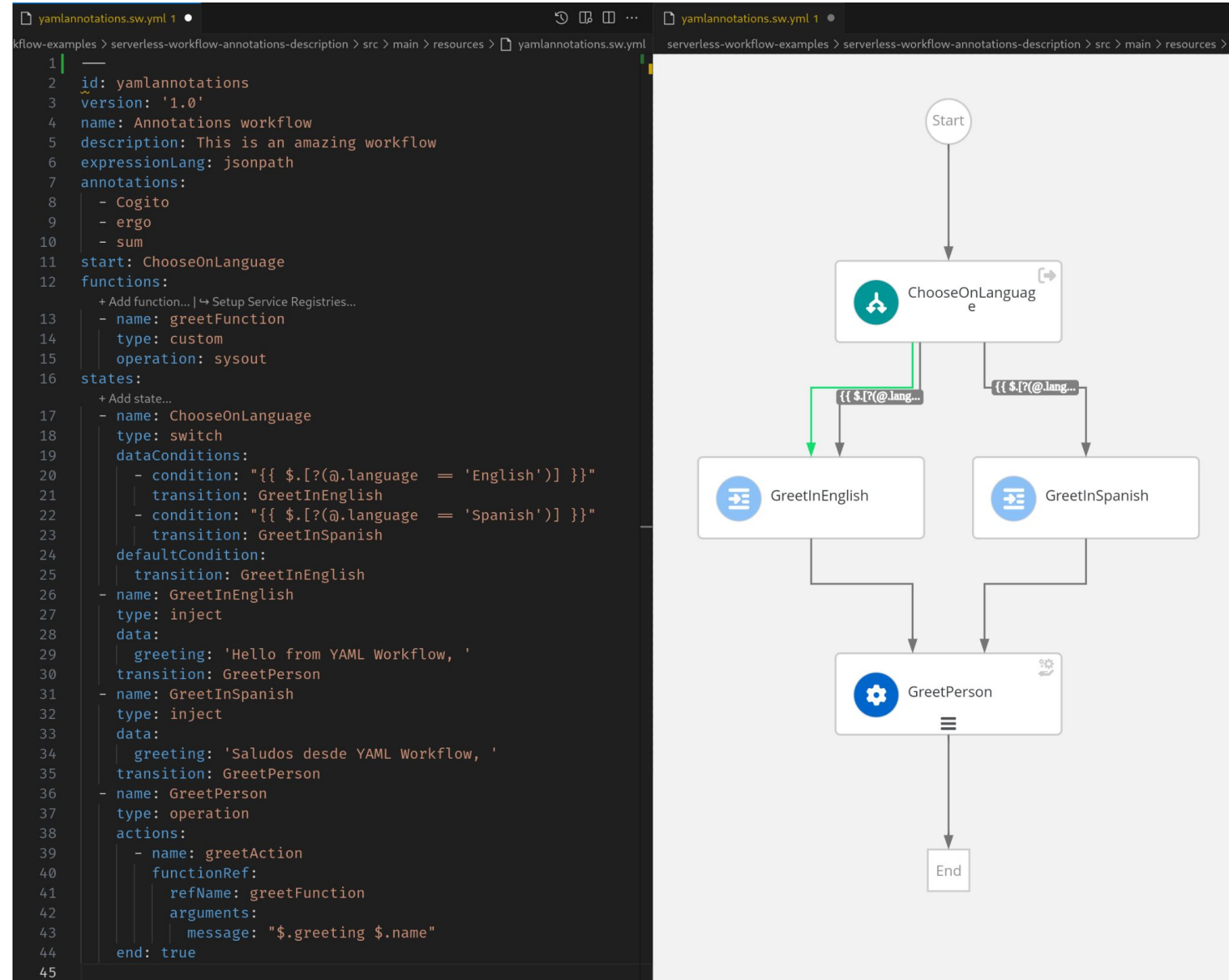
Available Types:

- Switch
- Inject
- Operation
- Event
- Sleep
- Parallel
- ForEach
- Callback

```
functions: []
states:
- name: ChooseOnLanguage
  type: switch
  dataConditions:
    - condition: "{{ $.[?(@.language == 'English')] }}"
      transition: GreetInEnglish
    - condition: "{{ $.[?(@.language == 'Spanish')] }}"
      transition: GreetInSpanish
  defaultCondition:
    transition: GreetInEnglish
- name: GreetInEnglish
  type: inject
  data:
    greeting: 'Hello from YAML Workflow, '
  transition: GreetPerson
- name: GreetInSpanish
  type: inject
  data:
    greeting: 'Saludos desde YAML Workflow, '
  transition: GreetPerson
- name: GreetPerson
  type: operation
  actions:
    - name: greetAction
      functionRef:
        refName: greetFunction
        arguments:
          message: "$.greeting $.name"
```

Serverless Workflow VSCode Extension

- Code Hints
- Code Snippets
- Diagram Generation
- Exporting Diagram to PNG



What is SonataFlow?



"**SonataFlow** is a tool for building cloud-native workflow applications. You can use it to do the services and events orchestration and choreography."

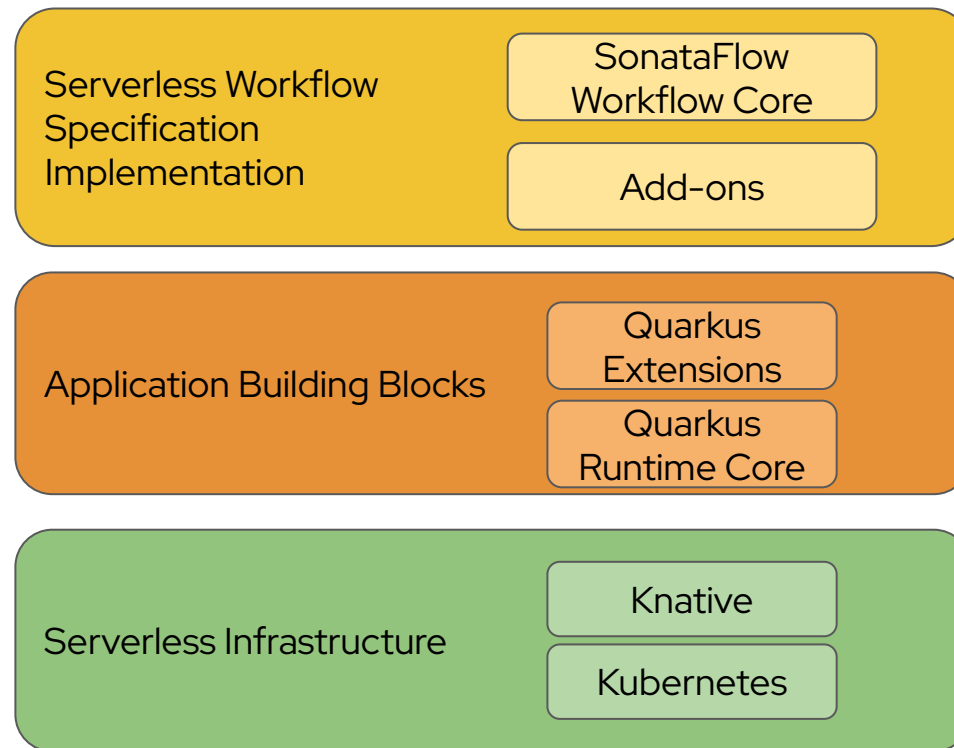
What is SonataFlow?



SonataFlow is the Red Hat implementation
of CNCF Serverless workflow
specifications

SonataFlow

Building blocks



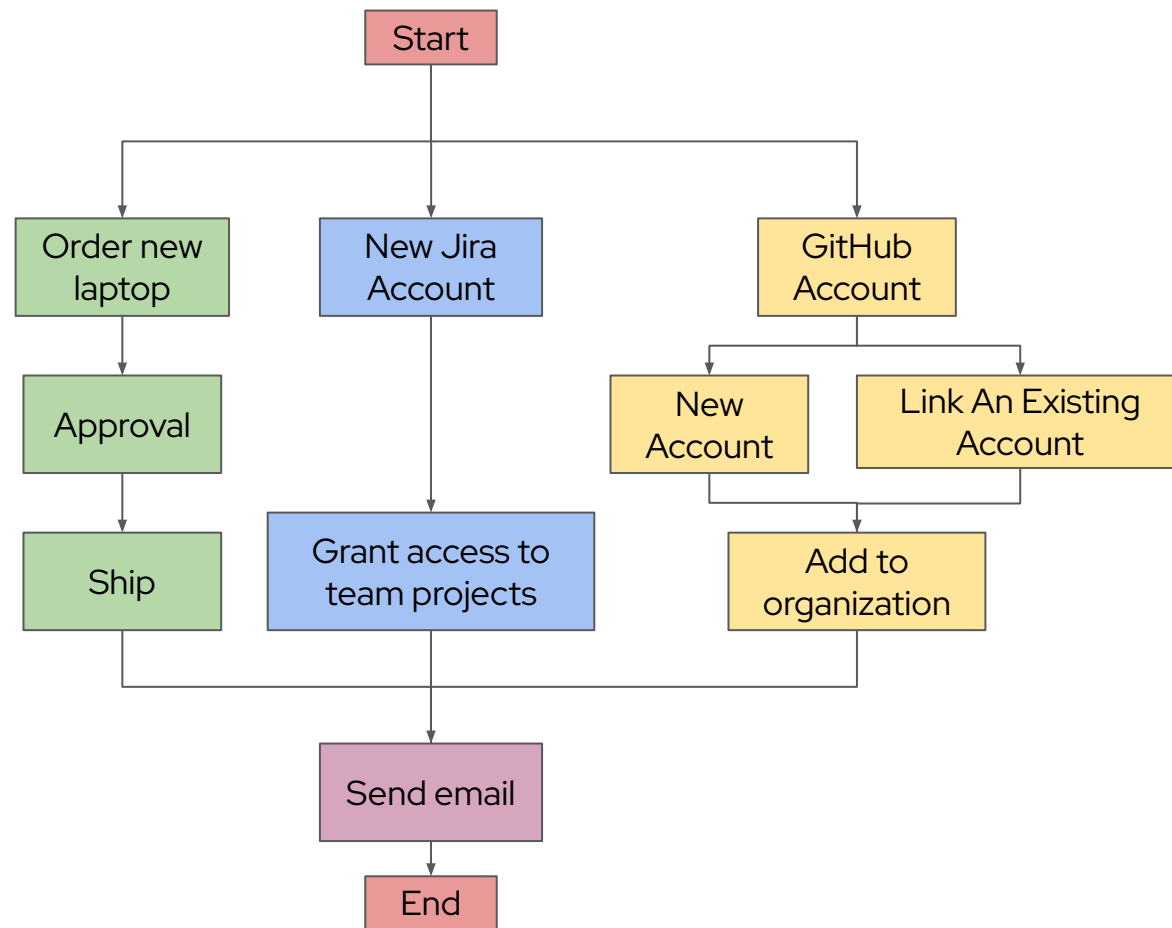
Give me use cases

Examples are available at

<https://github.com/kiegroup/kogito-examples/tree/main/serverless-workflow-examples>



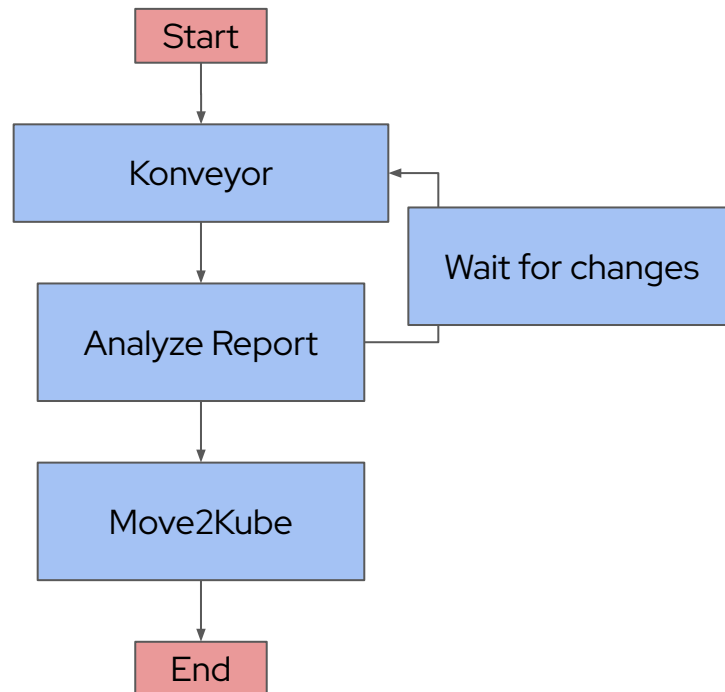
New Hire Onboarding



When a company hires a new employee, the IT department must:

- Set up all the necessary accounts
- Order and ship a new laptop
- Send a welcome email

Application modernization for the cloud



Analyze if an application can be migrated into the cloud and migrate it when ready

Demo time!



<https://github.com/gciavarrini/linuxday-2023>




Questions?

A vertical red bar on the left side of the slide contains a complex, stylized graphic. It features various icons: a cloud with a keyhole, a database cylinder, a server rack, a computer monitor, and several arrows pointing in different directions. There are also 'X' and 'O' symbols scattered throughout the design.

Thank you!

 <https://www.linkedin.com/in/gloria-ciavarrini/>

 <https://github.com/gciavarrini>

 gciavarrini@redhat.com