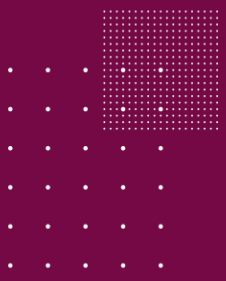
CANONICAL



Charm Developer

Training Curriculum

Prepared For:	Canonical
Prepared By:	Jon Seager
Prepared On:	22 June 2021
Version:	1.0

Introduction & Scope

This course is an introduction to charmed operator development and covers the development, testing and release of charmed operators for both machines and Kubernetes. The course is delivered as a blend of theory lessons and hands-on practical labs. Upon completion, attendees will feel comfortable developing charms that can manage applications throughout their life, and integrate seamlessly with other applications across computing substrates.

It is recommended that attendees are familiar with the basic concepts of Juju and the deployment/administration of charmed operators, either through experience or prior attendance to the Juju Administrator training course.

The duration of the course is two days.

Agenda

Day 1

Charmed Operator Developer Part 1

Agenda

Section	Notes	Time
Introduction	This section should be used to introduce the course material, explain the outcomes and what will be achieved in the labs - Verify prior knowledge/Juju Basics attendance - Verify that students have a suitable lab setup - Introduce Multipass for creating VMs quickly on physical hardware if relevant - Talk through installing Multipass, LXD, Juju, MicroK8s, Charmcraft as required - Outline the rest of the course	15 mins
Charmed Operator Anatomy	- What is an operator charm? - What is contained within a charm, directory structure - Discuss and demonstrate the contents of mandatory files (metadata.yaml, charm.py, etc) - Illustrate using charmcraft to initialise a new charm - Build the charm and illustrate that it is in fact just a zip file that can be inspected as such - Illustrate deploying the charm locally onto microk8s - Illustrate the output of `juju debug-log` and how to adjust the verbosity for the purposes of debugging	30 mins
Lab 1	- Ensure that MicroK8s is bootstrapped with a controller - Ensure that LXD is bootstrapped with a controller - Initialise a new charm with `charmcraft init` - Build the charm and deploy to MicroK8s	30 mins
Break	Break / Informal Q&A	10 min

Charmed Operator Framework Constructs	- Explain the constructs made available by the charmed operator framework
	  - StoredState
	  - Model
	  - Application/Unit
	  - Config
	- Explain the lifecycle events and some example event orderings
	- Demonstrate how to write logs using `logger`
	- Take questions on any of the above
		45 mins

Charmed Operator Developer Part 2

Agenda

Section	Notes	Time
Starting a workload	Begin with how to start a workload on Kubernetes - Detailed introduction to Pebble and it's configuration - Walkthrough the charm template example with httpbin - Explain the possible Pebble layer config options - Adjust the template to start a different container from the template (something simple like nginx or similar that will serve up something obvious with no config) - Deploy the charm - Inspect the deployment and show the two containers in the pod with `kubectl` Move onto starting a workload in a machine charm - Create a new charm using `charmcraft init` - Strip out the Kubernetes specific stuff (containers/resources map in metadata and the pebble_ready handler) - Talk through starting the same workload as used in the previous example ideally - Installing packages in the install method, starting services with systemd in start method etc. Introduce the concept of application status and what the various states (Active, Maintenance, Blocked, Waiting) mean and when to use them - Deploy the machine charm to LXD and verify the application is working correctly - Take questions on any of the above	1 hr 30 min
Lab 2	- Create and deploy a Kubernetes charm that deploys nginx with no config - Validate the site is accessible by reaching the pod IP on port 80 - Create a machine charm that deploys nginx with no config on LXD - Validate the site is accessible by reaching the pod IP on port 80	45 mins
Break	Break / Informal Q&A	10 mins
Handling Configuration	- Explain where config items are defined and how they are accessed (read-only) from the charm code - Explain the config.yaml schema and principles behind designing "good" config for charms - Implement a config option in the Kubernetes charm and explain the config-changed handler - Demonstrate how to access config items from within the code - Outline the following lab and the steps that will need to be taken to complete it	45 mins
Lab 3	- Adjust the LXD nginx charm so that it can take a port for nginx to listen on - Implement the config-changed handler so that the nginx config is adjusted based on the config value - Deploy the charm and test that adjusting the config value has the desired effect	30 mins

Working with Containers	- Explain how Pebble enables the author to
	  - Push/pull files
	  - List files
		45 mins

	- Make/remove files and directories - Walk through an example of push/pull a file - Signal the location of the docs online - Demonstrate how to get access to a container using `self.unit.get_container()` - Explain the following lab	
Break	Break / Informal Q&A	10 mins
Lab 4	- Adjust the Kubernetes charm so that it takes the same config option for nginx (listen port) - Implement the config-changed handler so that when the config is changed, the charm pushes a new nginx config file into the container using the Pebble api and restarts nginx	45 mins
Actions	- Briefly recap the purpose and utility of charm actions - Give some examples of actions - deploy and charm and demonstrate invoking a charm action and how to get the results and logs - Introduce the actions.yaml file and talk through a couple of examples - Demonstrate how to implement a simple action by defining it in actions.yaml and then implementing the handler	30 mins

Day 2

Charmed Operator Developer Part 3

Agenda

Section	Notes	Time
Lab 5	- Instructor should have prepared a simple web page accessible at a public URL (pastebin/gist/similar) - Define an action to pull a new website that takes a URL as a parameter - Implement the action handler so that it fetches the HTML file from the given URL and puts it into the webroot for the nginx charm on LXD or MicroK8s (or both as an extension) - Verify that running the action with different URLs results in nginx serving up different pages	45 mins
Relations	- Recap how relations and relation data bags work - Explain relation interfaces - Introduce the metadata.yaml syntax for defining provides/requires relations and peer relations - Walkthrough at least one simple example of how relations are defined and handled using example charms from charmhub - Explain the differences between the different relation events and when they are triggered - Demonstrate how relation data is set at unit/application level	1 hr
Lab 6	This lab requires the instructor to have access to a basic charm that starts a workload expecting a relation to a database (mariadb, for example) - The charm should be preconfigured to start and wait for a relation to be defined - Have skeleton methods for relation_joined and relation_changed events but no implementation - The application started should be a hello world app written in python that connects to the database, inserts some data then displays it on a webpage (gunicorn charm?) Student should then: - Deploy the supporting database charm - Implement the relation_joined and relation_changed methods so that the application can function	1 hr 30 min
Break	Break / Informal Q&A	10 mins
Libraries	- Explain the purpose of charm libraries - Explore some example libraries on charmhub - Demonstrate fetching/listing libraries with charmcraft - Common use cases for libraries - relation interfaces	30 mins
Lab 7	- Implement a relation to the ingress controller on MicroK8s using the `nginx-ingress-integrator` charm library for the Kubernetes charm	30 mins

Testing	- Introduce the framework "Harness" - Explain the constructs it provides for simulating deployed charms - Explain those methods that are specific to Kubernetes charms (like the pebble_ready methods) - Talk through some tests in an existing charm - Talk more broadly about test coverage and adding functional tests with a framework like zaza	30 mins
Break	Break / Informal Q&A	10 mins

Lab 8	- Implement unit tests for the Kubernetes nginx charm to test
	  - Pebble config
	  - File push/pull
	  - Config
	  - Action
	  - Start/stop of the nginx process
	- Take questions until students finish the task
		1 hr

Publishing and Documentation	- Outline the publishing process - Uploading -> Uploading resources -> Release and associate resources - Explain the `charmcraft` commands required - Explain the charm documentation system using Discourse - Demonstrate setting up charm documentation with an invisible topic in discourse, link in metadata.yaml - Explain how to add icons and other metadata like to homepage to a charmhub listing	45 mins
Lab 9	- Choose between the Kubernetes and LXD charm - Upload the charm, adjusting the metadata to set the name to `<alias>-nginx` or similar to prevent clashes - Release the charm into the edge channel - Create a charm documentation topic in discourse - Create a new release of the charm including the documentation and verify it is showing correctly - Release the charm into stable - Add an icon and homepage to the charmhub listing	45 mins