DiracX

Deployment

Deploying DIRAC

- runit - Build our own RPM as it’s no longer maintained
- Custom install scripts dirac-install-* for each part
- Custom install.cfg file
- Additional steps needed to make the installation work
- Updates managed completely separately (SystemAdministrator)
- Each handler is a different process (except Tornado)

Deploying DIRAC

- runit - Build our own RPM as it's no
- Custom install scripts dira part
- Custom install.cfg f
- Additional the installation work
- Upd tely separately (SystemAdministrator)
- s a different process (except Tornado)

BUT "easy and well documented"
Igor, 16.10.2023

In principle, it will all be better

What does everyone else do these days?

- Kubernetes - Standard to define a distributed system
- Separate infrastructure from applications
  - “Please IT department(/cloud provider) run this for me”
- Helm gives the ability:
  - to parameterise
  - distribute a kubernetes config

How will we distribute DiracX?

- We'll provide a helm chart: https://github.com/DIRACGrid/diracx-charts

- How can you use it?
  - If your institution provides a kubernetes* service: use it
  - If you work with public clouds: use their container services
  - If you're a smaller install: use a lightweight option k3s/k0s/rke2
    - We'll give a recommendation before releasing DIRAC v8.1
    - This recommendation will be used for the DIRAC Certification
    - If you already have experience with these, let us know!

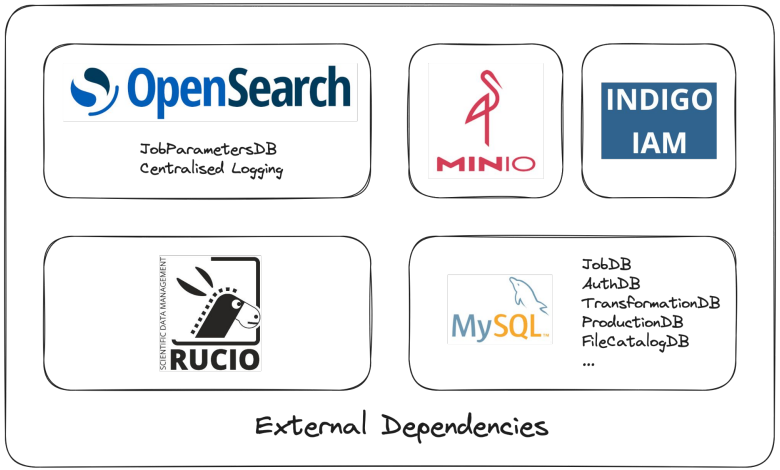
*Also packaged as a service:

What does the helm chart contain?

- At a high level there are three things you need to run

What about dependencies?

- Helm lets us also depend on other charts



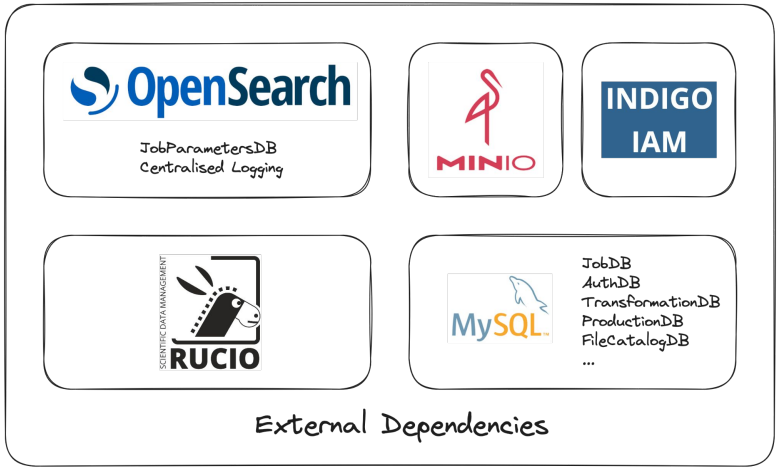
What about dependencies?

- Helm lets us also depend on other charts

https://opensearch-project.github.io/helm-charts/

https://charts.bitnami.com/bitnami/

https://charts.min.io/



https://github.com/DIRACGrid/diracx-charts

How was DiracX installed for DIRAC-Certification?

- YAML configuration file to choose what you want to run

```yaml
diracx:
  hostname: diracx-cert.app.cern.ch
  settings:
    DIRACX_SERVICE_AUTH_ALLOWED_REDIRECTS: '["https://diracx-cert.app.cern.ch/api
    DIRACX_DB_URL_AUTHDB: "mysql+aiomysql://.../DiracXAuthDB"
    DIRACX_DB_URL_JOBDB: "mysql+aiomysql://.../JobDB"
    DIRACX_DB_URL_JOBLOGGINGDB: "mysql+aiomysql://.../JobLoggingDB"
    DIRACX_DB_URL_SANDBOXMETADATADB: "mysql+aiomysql://.../SandboxMetadataDB"
    DIRACX_LEGACY_EXCHANGE_HASHED_API_KEY: xxxxxxxxx
    DIRACX_SANDBOX_STORE_BUCKET_NAME: diracx-cert-sandboxes
    DIRACX_SANDBOX_STORE_AUTO_CREATE_BUCKET: "true"

minio:
  enabled: true

mysql:
  enabled: false

opensearch:
  enabled: false
```

The values.yaml can include a lot more

- When DiracX is complete, you'll be able to run the full system including all its dependencies
  - Can even include example compute elements, storage elements?

- This is used for:
  - DiracX testing (GitHub actions)
  - Local development instance
  - Running a demo instance
  - Running the DIRAC certification instances
  - Running your production instances

How will the first installation of DIRAC v8.1 look

- We'll provide a script to generate the initial values.yaml from your existing config
  - Ongoing https://github.com/DIRACGrid/diracx/pull/140
  - Few values will need editing

- You'll then be able to enable/disable DiracX traffic
  - via configuration flags in the DIRAC CS

Running a demo/local developer instance?

- In the diracx-charts repo we have “./run_demo.sh”
- Please **don’t** run it now (it downloads a lot of data)
- This (should) work on any machine with docker
  - No need for sudo or other dependencies
  - A few known limitations at the moment - solution are known

Runni

- In
- Ple
- Thi
  -
  -

```
🚧 Chart has been installed in developer mode.

ℹ️ To interact with the cluster directly using kubectl/helm you'll need to set the
first following environment variables:

  export KUBECONFIG=/Users/cburr/Development/DIRAC-future/diracx-charts/.demo/kube.conf
  export HELM_DATA_HOME=/Users/cburr/Development/DIRAC-future/diracx-charts/.demo/helm_data
  export PATH=${PATH}:/Users/cburr/Development/DIRAC-future/diracx-charts/.demo

Then see the chart README for more information on how to use kubectl/helm.

ℹ️ To use the demo with the DiracX client set:

  export DIRACX_URL=https://christophers-macbook-pro-5.local:8000
  export DIRACX_CA_PATH=/Users/cburr/Development/DIRAC-future/diracx-charts/.demo/demo-ca.pem

ℹ️ To access the web application visit:

  https://christophers-macbook-pro-5.local:8000

ℹ️ To access the interactive API documentation visit:

  https://christophers-macbook-pro-5.local:8000/api/docs

ℹ️ When prompted to login the credentials are:

  Username: admin@example.com
  Password: password
🦄 Waiting for installation to finish...
pod/diracx-demo-7d9f94fcb6-zn8kp condition met
🎉 🎉 🎉 Pods are ready! 🎉 🎉 🎉

ℹ️ Press Ctrl+C to clean up and exit
```

Questions?