

2023 2Q update

Requirements & Constraints

- Security/secrets not directly addressed - scope of other discussions, but relevant here - especially exposed via environment/files
- Beyond config/repositories (stored externally via connectors/standard k8s objects), Only ephemeral storage - otherwise immutable. Declarative
- One egeria server per pod
- Aim for early results, build on what exists
- top level server definition/type in k8s object, but most config remains in json document(s)
- Integration use-case
- Multiple stages: assembling a container (capabilities) vs deploying

Progress

- [Education](#) on eclipse microprofile
- Identified [list of experiments/investigations](#)
- Example application which runs only a single server (no platform configuration). See [PR](#) . Addresses biggest pain point of having to run rest API calls to configure egeria server after container started
- [Repo](#) created for experiments

Other recent achievements in k8s charts/containers

- More connectors have an associated docker container, built using the same version of Egeria as that the connector pre-reqs
- As part of Java 17/Egeria 4 work, containers are now using a Java 17 UBI-9 containers, which also now contains more tools (sdk, not just runtime) to facilitate monitoring/debugging - BUT we need to understand more around sizing (memory, limits, k8s...)
- Atruvia's contributions to helm charts are merged which provides more configurability on the base chart (such as Kafka endpoints), and provision of a new lineage chart (which still needs some fixes)
- For lab charts, storage class now configurable
- 'snippet' samples for helm charts

Other work in progress/to be done

- Memory footprint - currently JVMs may grow until container environment kills them. Support added to limit resources, but default remains unlimited as unable to get a stable environment

Next Steps