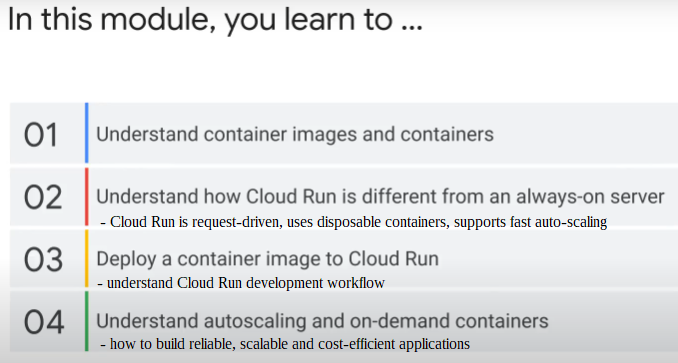
Application Development with Cloud Run

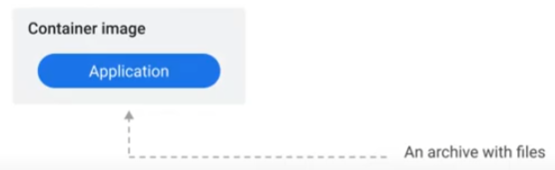
### Understanding Cloud Run

#### Introduction

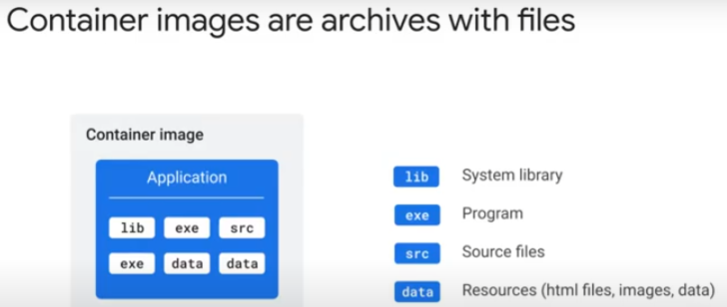


### Introduction to containers

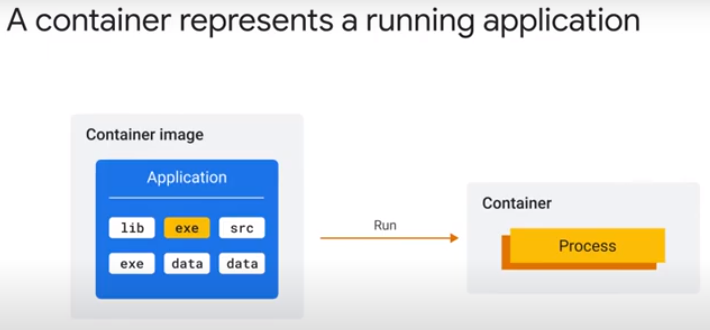
Structure of container image



* a package of your application, plus everything else needed by it to run
* e.g. Java application: application + Java virtual machine

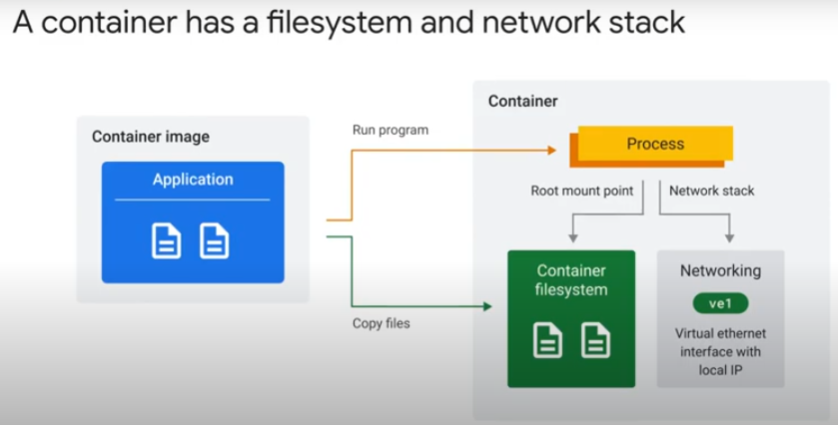


* an archive of files: source codes, executable programs, system libraries, resources (HTML files, images, binary blobs, dependencies)
* package can be stored, downloaded and sent somewhere else
* program can be written in any programming language, e.g. Java, Javascript, Python, PHP, Go.

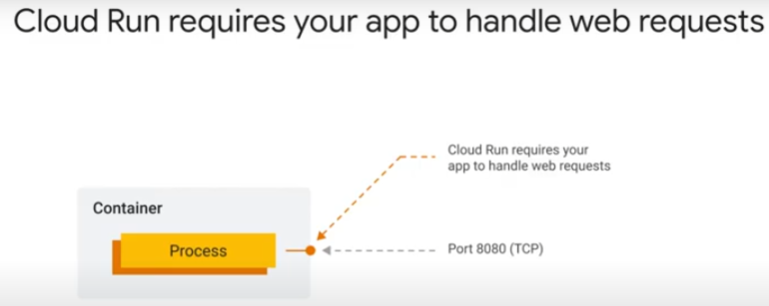


Running a container

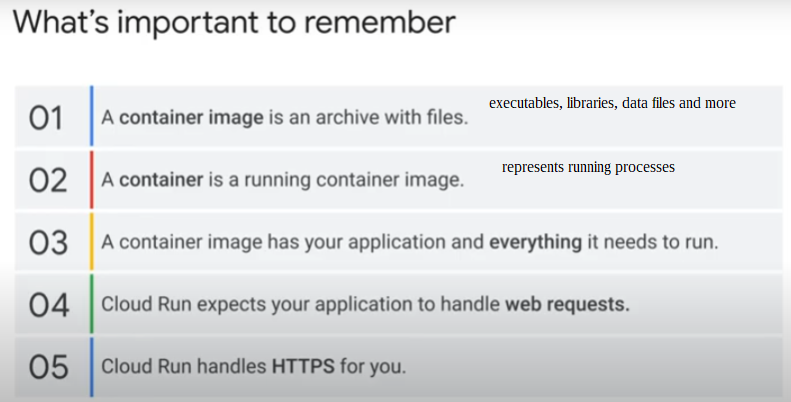
* means executing program inside container image
* E.g., if Java application in image > execute Java Virtual Machine (JVM) when run container
  + if node.js application, source files and node.js runtime are in the container image
* container represents the running processes of the application
  + no running processes > no container
  + container only exists at runtime



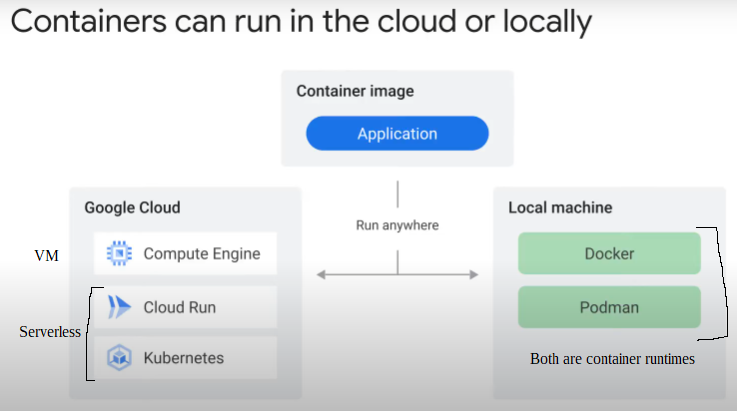
* A private file system for container created from the content of the container image
  + processes of application see these files
* processes have access to a private virtual network interface with a local IP
  + application can bind to interface and listen to a port for incoming traffic
  + a process can acces its own private virtual network stack > open a port > and listen for incoming requests
  + port 8080 is configurable default
    - if unavailable, change configuration in application to use another port
  + Cloud Run expects/requires application to handle web requests

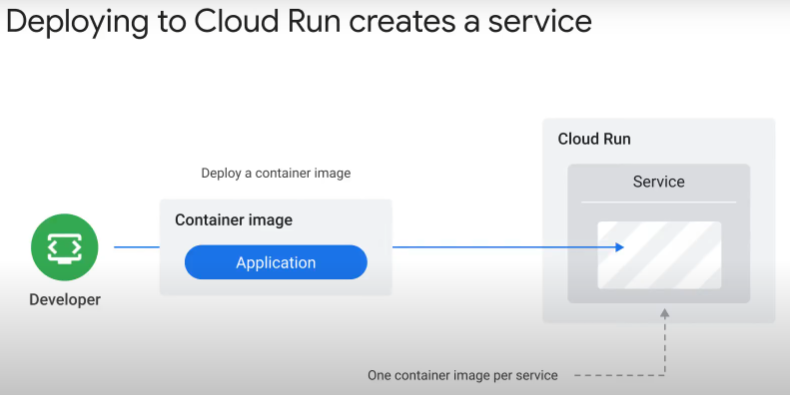


* no need to provide HTTPS server. Provided by Cloud Run/Google Cloud infrastructure.

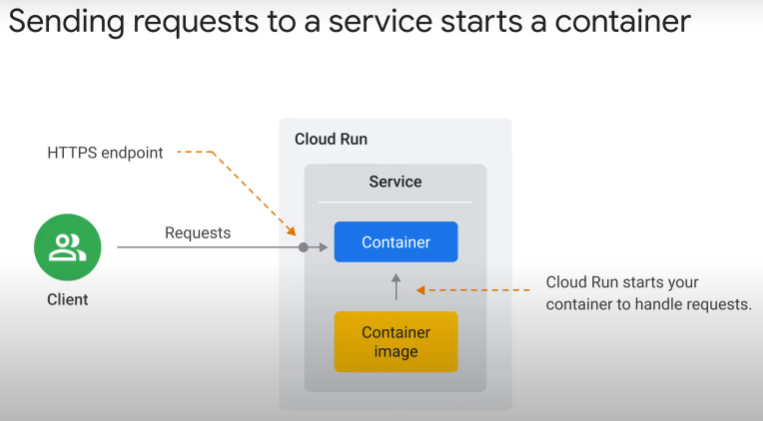


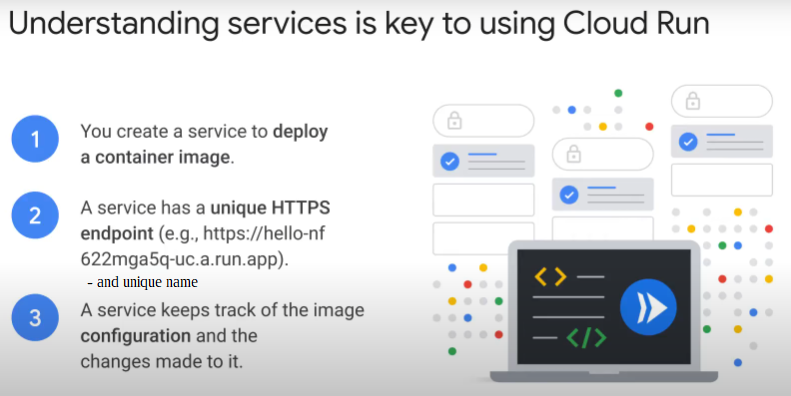
### Running your application on Cloud Run





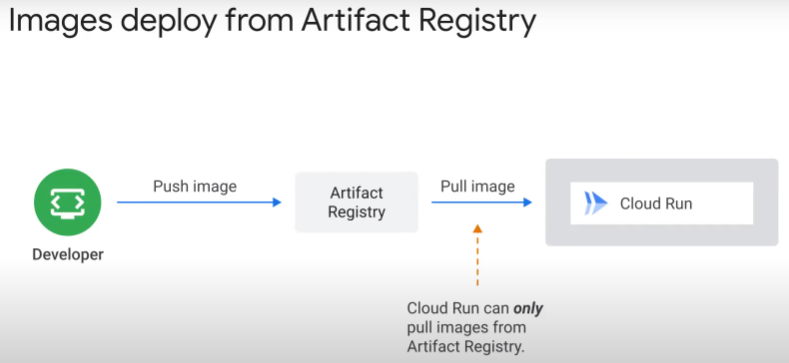
* When dev deploy container image to Cloud Run, Cloud Run creates a Service.
* A service contains only 1 container image
  + each service has unique HTTPS endpoint on a sub-domain of the run.app domain
  + service forwards incoming requests to container/s
  + initially, when there are no containers, Cloud Run service starts a container to handle the requests
  + service will dynamically add and remove containers to handle all incoming requests
    - auto-scaling
  + incoming traffic can also be routed through a global HTTPS load balancer

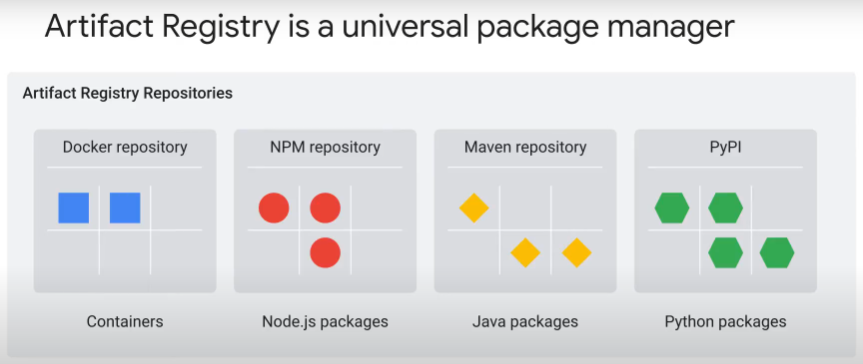


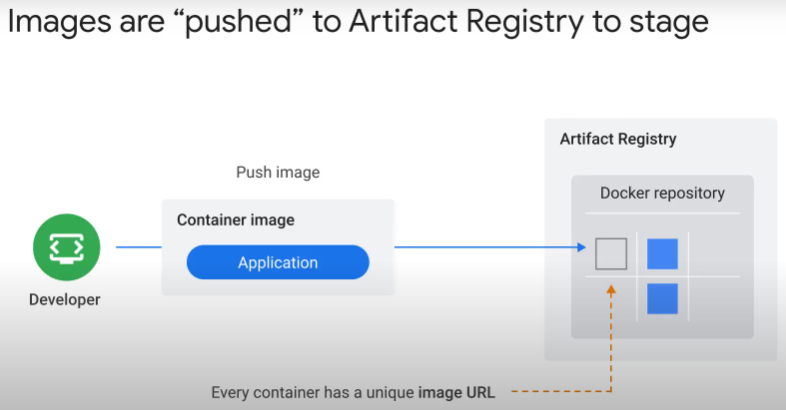
* dev interaction is mainly with service for many tasks, e.g.
  + deployment
  + roll-back
  + update configuration settings, e.g. environment variables
  + set/change scaling boundaries

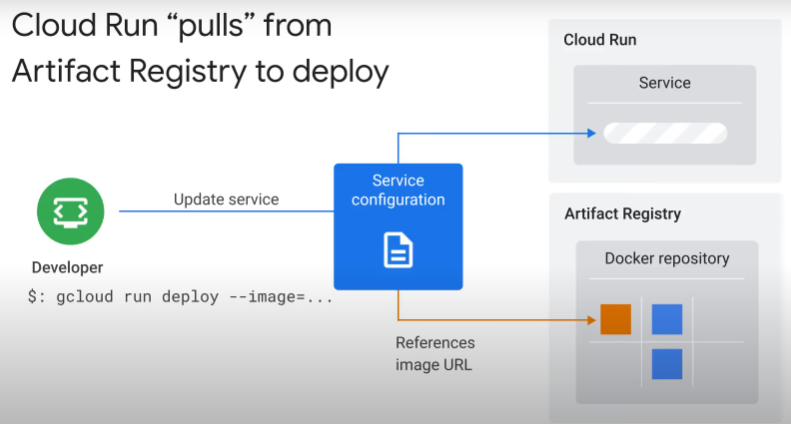
How container image gets to Cloud Run

* When deploy container image, it is sent to Artifact Registry as intermediate storage location. Not to Cloud Run directly.
* Cloud Run can **ONLY** pull images from Artifact Registry

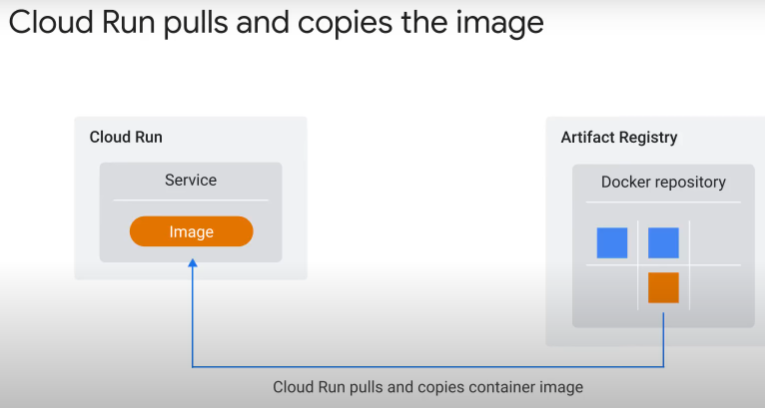




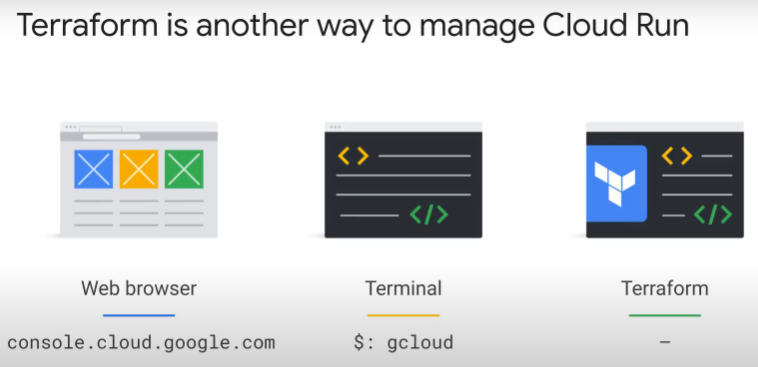
* Artifact Registry is a fully managed service previously known as Google Container Registry (GCR).
  + Single place for container images and language packages (Maven, npm)
    - universal package manager
    - host container images privately
  + create a Docker repository to push container images to and pull from
  + every image will have **unique** name and url in repository



* “gcloud run deploy –image=.... “ passes container reference to Cloud Run
  + Cloud Run pulls (downloads) image from Docker repository
    - stores copy of image locally
    - in container’s internal RAM storage, which is fast
    - ensure service starts reliably and quickly
    - container storage has a default size. Pls check that large image size is not an obstruction
    - because Cloud Run copies the image to the container’s storage, Cloud Run service will not be affected if the deployed image in Artifact Registry is deleted



How to manage Cloud Run



3 ways:

* Google Cloud Console
  + have guides and wizards
* Cloud Shell terminal. gcloud command line.
  + can write scripts (.sh) for automation.
  + gives helpful suggestions if mistype or error.
* Terraform
  + declarative.
  + For provisioning and maintenance of Cloud infrastructure
  + describes all resources and links between them in yaml file.
    - Terraform builds based on this specification/manifest file.
  + Infrastructure-as-Code (IaC)
  + helps build consistent and repreducible environments across teams.