Container Environment

This page describes the resources available to Containers in the Container environment.

Container environment

The Kubernetes Container environment provides several important resources to Containers:

- A filesystem, which is a combination of an image and one or more volumes.
- Information about the Container itself.
- Information about other objects in the cluster.

Container information

The *hostname* of a Container is the name of the Pod in which the Container is running. It is available through the `hostname` command or the `gethostname` function call in libc.

The Pod name and namespace are available as environment variables through the downward API.

User defined environment variables from the Pod definition are also available to the Container, as are any environment variables specified statically in the Docker image.

Cluster information

A list of all services that were running when a Container was created is available to that Container as environment variables. This list is limited to services within the same namespace as the new Container's Pod and Kubernetes control plane services. Those environment variables match the syntax of Docker links.

For a service named *foo* that maps to a Container named *bar*, the following variables are defined:

```
FOO_SERVICE_HOST=<the host the service is running on>
FOO_SERVICE_PORT=<the port the service is running on>
```

Services have dedicated IP addresses and are available to the Container via DNS, if DNS addon is enabled.

What's next

- Learn more about Container lifecycle hooks.
- Get hands-on experience attaching handlers to Container lifecycle events.

Feedback

Was this page helpful?

Yes No

Last modified August 10, 2021 at 11:15 AM PST : Hard-code the name of the target repo's default branch instead of using the githubbranch parameter value (f945335af)