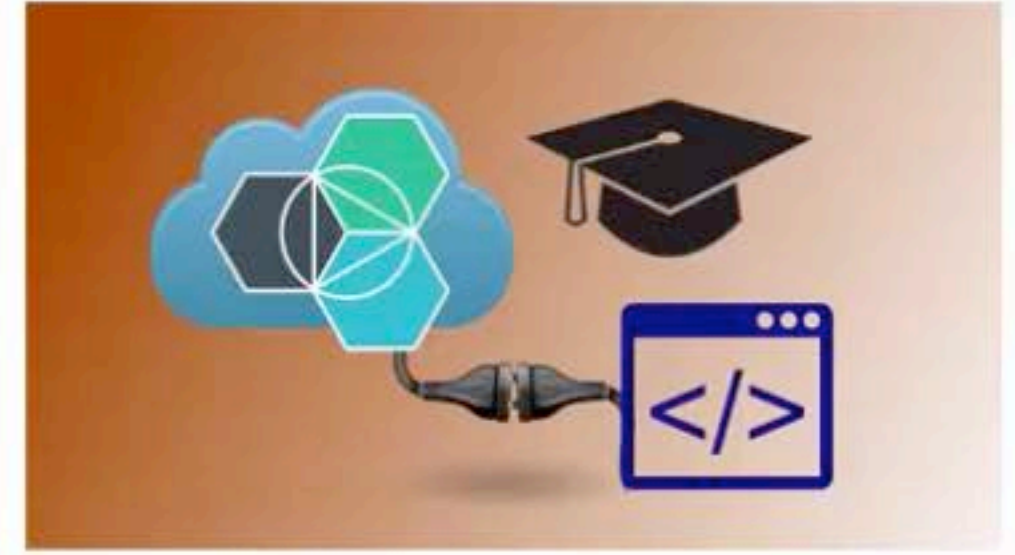


# Environment Variables



## Learning Objectives:

1. Understand the environment variables available to apps
2. Learn how to use user defined environment variable

# CF Environment Variables

- Means by which CF runtime communicates with the application
- Environment variables are set by various CF components such as DEA, Buildpacks
- Developers can set application specific environment variables – referred to as user defined environment variables

# Environment variables - Categories

Application

- Common to all instances of the app

Instance specific

- May be different for each instance of app

User Provided

- User provided environment variables

Environment Variable  
Groups

- Common across all apps/system

# Application

- Common to all instances of the app - set by DEA & Buildpack during staging
  - HOME
  - PORT
  - TMPDIR
  - VCAP\_APP\_HOST
  - ~~VCAP\_APP\_PORT~~
  - MEMORY\_LIMIT
  - PWD
  - USER
  - VCAP\_APPLICATION
  - VCAP\_SERVICES
- Buildpack may also specific variables e.g., JAVA\_HOME



# CLI Commands for environment

```
cf env APP_NAME
```

Shows all environment variable for the app

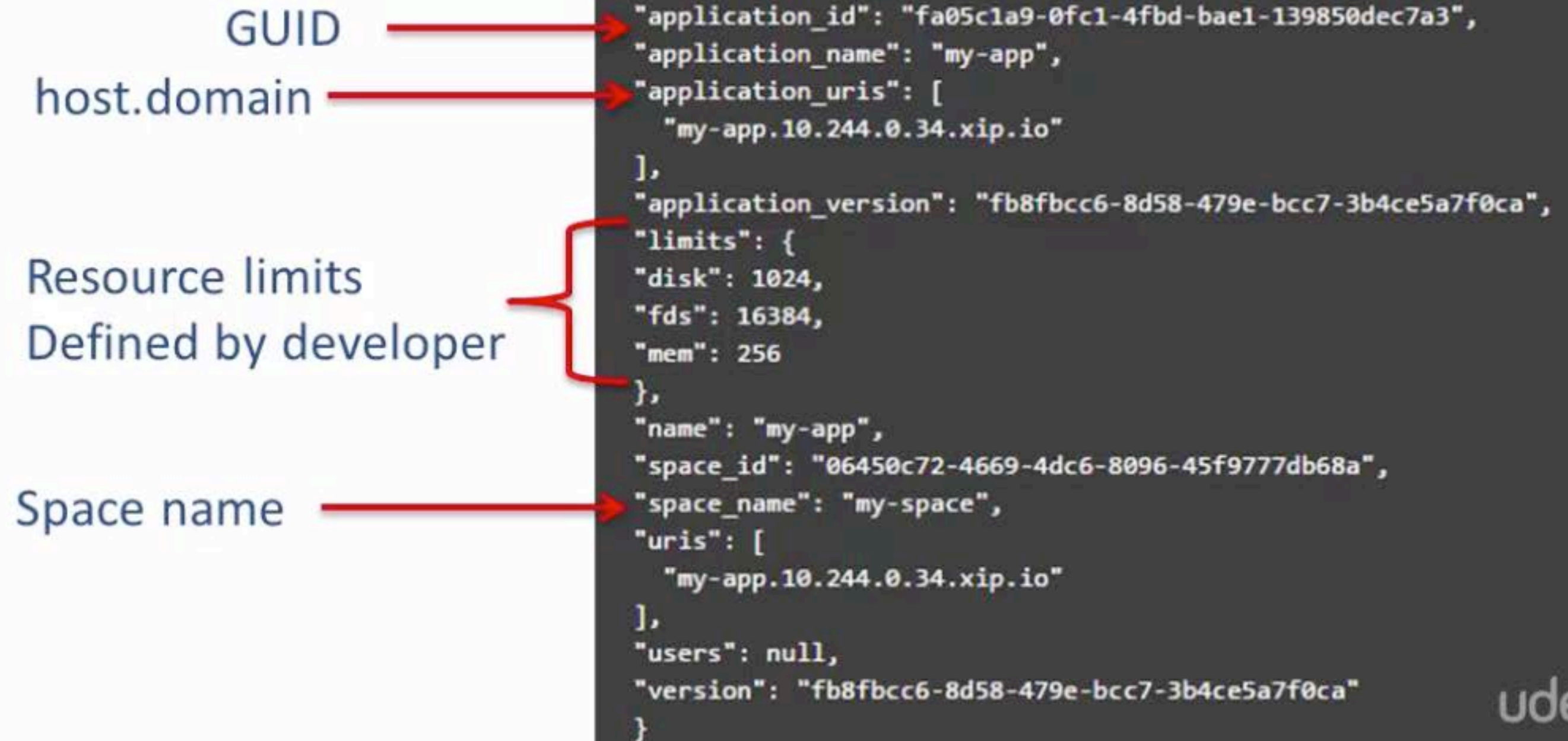
```
cf set-env APP_NAME ENV_VAR_NAME ENV_VAR_VALUE
```

Sets the user provided variables. Can also be set using MANIFEST

# VCAP\_APPLICATION

VCAP = VMWare's Cloud Application Platform

- In JSON format



# VCAP\_SERVICES

- For bindable services CF adds details to the VCAP\_SERVICES in JSON format

Credentials  
For the service

Name of service

Instance Name

Plan for service

```
"VCAP_SERVICES": {  
  "cleardb": [  
    {  
      "credentials": {  
        "hostname": "us-cdbr-iron-east-03.cleardb.net",  
        "jdbcUrl": "jdbc:mysql://us-cdbr-iron-east-03.cleardb.net:3306/  
        "name": "ad_6e^4^a--a^80b2",  
        "password": "c xxxxxxxxx",  
        "port": "3306",  
        "uri": "mysql://^60eb6:c58c5b97@us-cdbr-iron-east-03.cleardb.net:3306/  
        "username": "he5 xxxxxxxxx :b6"  
      },  
      "label": "cleardb",  
      "name": "MySQL",  
      "plan": "spark",  
      "tags": [  
        "Data Stores",  
        "data_management",  
        "ibm_third_party"  
      ]  
    }  
  ]  
}
```



# Instance specific Env. Variables

- May be different for each instance of the same application
  - CF\_INSTANCE\_INDEX
  - CF\_INSTANCE\_IP
  - CF\_INSTANCE\_PORT
  - CF\_INSTANCE\_ADDR
  - CF\_INSTANCE\_PORTS



# User Provided Env. Variables

- Application specific environment variable
- Can be set via manifest file e.g., application\_stage=PROD

```
cf set-env APP_NAME VAR_NAME VAR_VALUE
```

Sets environment variable for the application

```
cf unset-env APP_NAME VAR_NAME
```

UnSets environment variable for the application

# Environment Variable Groups

- System-wide variables that enable operators to apply a group of environment variables to all running applications and all staging applications separately
- An environment variable group consists of a single hash of name-value pairs. Example is HTTP Proxy. Only the Cloud Foundry operator can set the hash value for each group

`cf revg`  
Retrieves the env variable group

`cf ssevg NV_PAIRS_JSON`  
Sets the env variable group

# Summary

- CF uses environment variables to pass information to the application
- User defined environment variables can be created using the CLI and manifest file