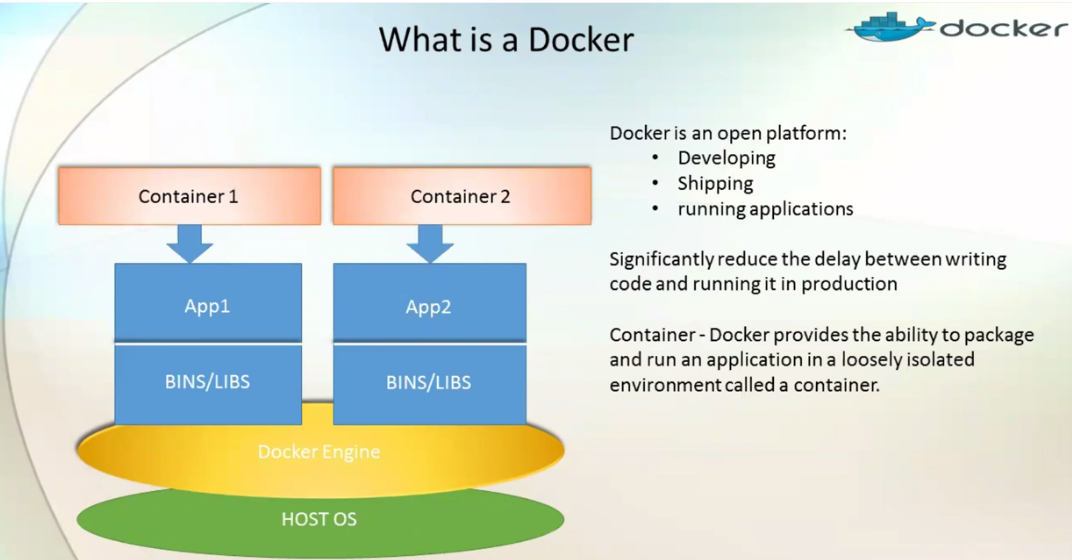
1. What is Docker?

Open source platform to

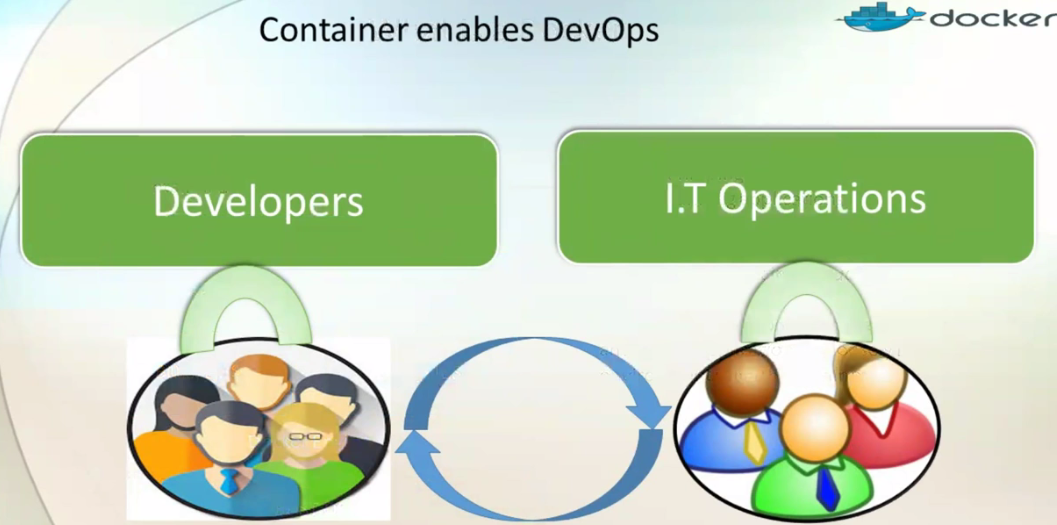
* 1. Deploy the code.
  2. Ship the code.
  3. And runt the app in production environment.

In the above diagram, we have **Host Operation System**. On top of which, **Docker Engine** is running. Docker allows us to create a no of **containers**. Container is not complete OS. OS is with Host OS.

The container has required

* 1. Binaries,
  2. Libraries  
     Which supports the app.

***For example***: If we need 4 containers it doesn’t mean that we need to install 4 Os on 4 containers. One container holds one application and there is isolation b/w the applications running in different containers. One doesn’t affect other app.

1. *Do*cker enables **DevOps.  
   For example:** We have developers (Write code) and IT Operations (Deploy the code). There is development life cycle 🡺 Write🡺Build🡺Test🡺Release

But b/w these stages, there is long time because

* 1. Sometimes when pre-development environment is not same as development environment.
  2. Stage and production have different versions of tools.

NOTE: This complication leads to long time to release the code into the production environment.

🡺So, here DevOps allows us to reduce that gap and provides a complete bridge b/w the developers and IT Operations.

🡺**What now the question is how we achieve this lesser time**?

🡺Due the the DevOps concept in Docker.

🡺All the tools we’re using in pre-productions, are created as a single unit (as container) and that single container can be used to create image and that image can be deploy into the production. So, there is no changing b/w the stages and production.

🡺So, same set of binaries, libraries, tools and app would be readily available in the production.