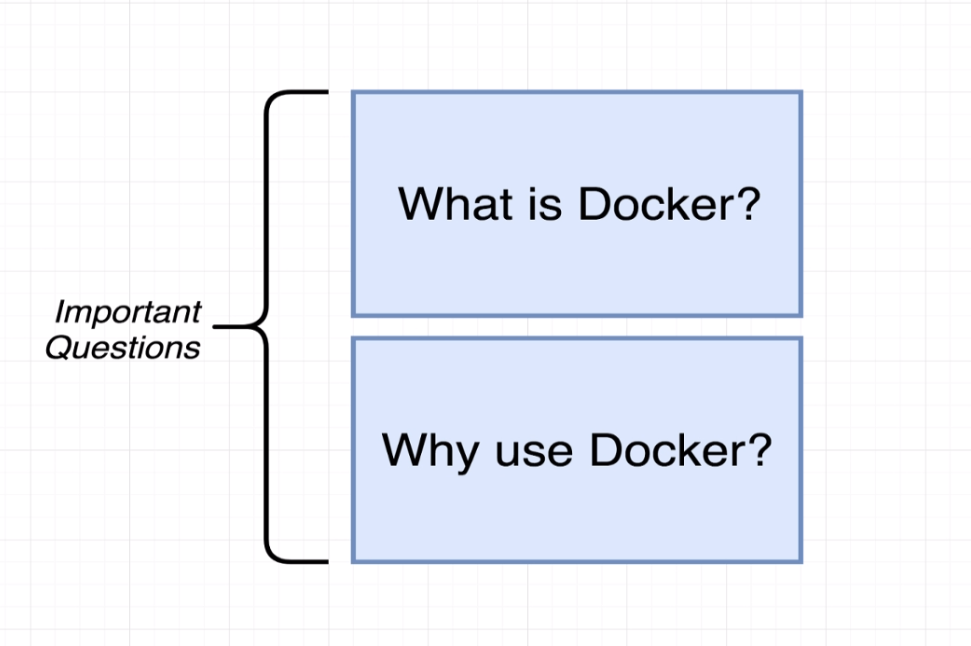
**Why Docker:**



Flow for installing software in your personal computer:

Diagram

Description automatically generated

Example: to install a software “Redis”:

Graphical user interface, text

Description automatically generated

Graphical user interface, text, application, email

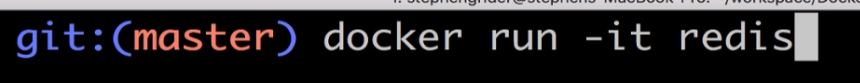
Description automatically generated

Text

Description automatically generated

Endless cycle of troubleshooting.

If we use docker, need to run one single command and instance of software will be installed in our computer:

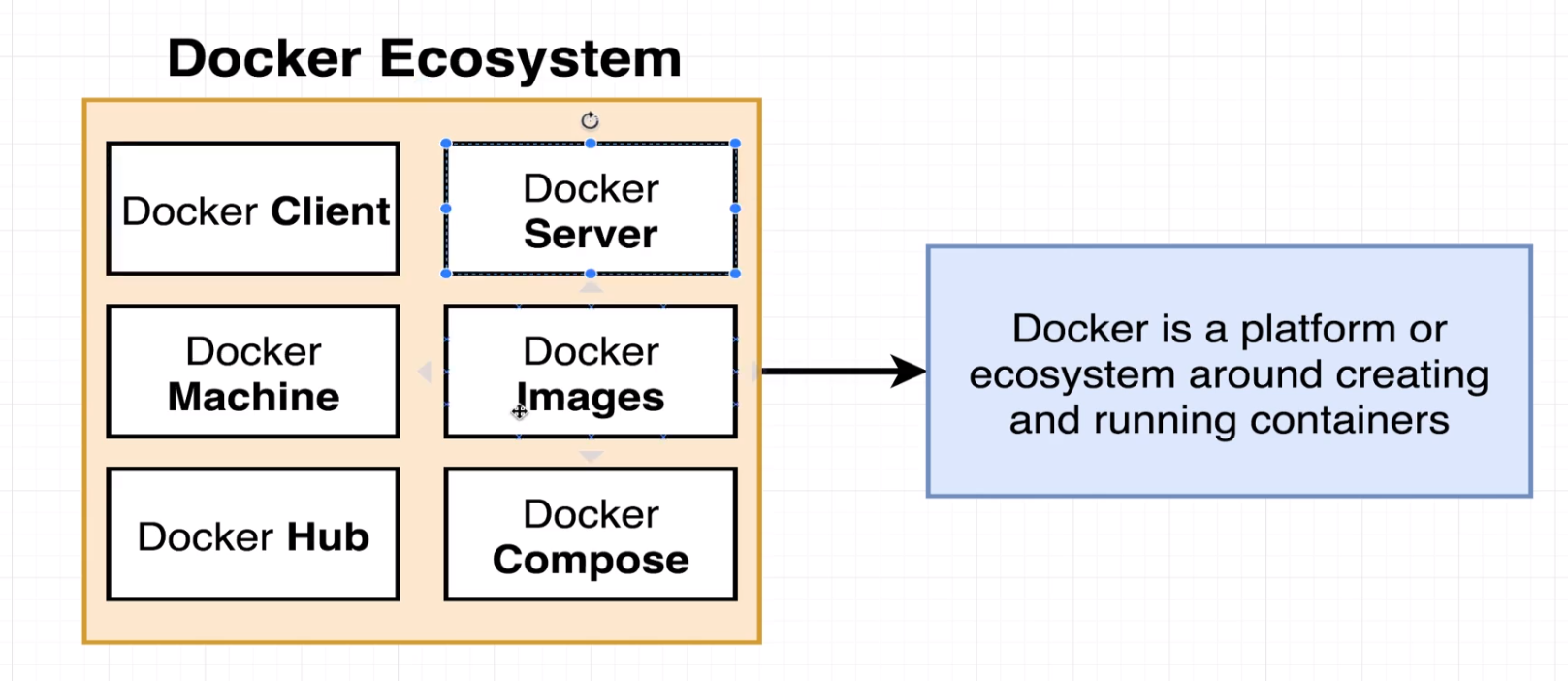


* When we ran above command at my terminal of “*docker run -it redis”*, it went through a little series of actions behind the scenes and we're going to examine that entire series of actions very closely over time.

Graphical user interface, application, table, Excel

Description automatically generated

**Docker** **ecosystem of different projects, tools, and pieces of software.**



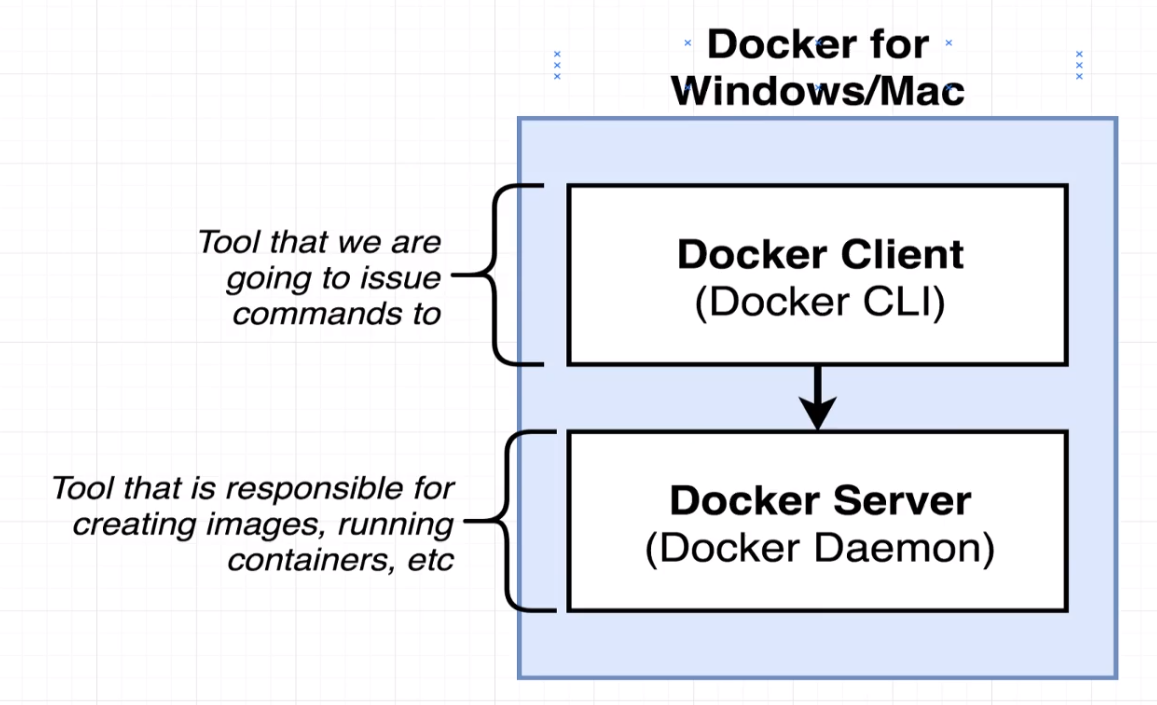
**IMAGE & CONTAINER**

* When we ran “*docker run -it redis”* command, something called the Docker CLI reached out to something called the Docker hub and it downloaded a single file called an image.
* An image is a single file containing all the dependencies and all the configuration required to run a very specific program.
* For example, Redis, which is what the image that we just downloaded was supposed to run. This is a single file that gets stored on your hard drive. And at some point, in time, you can use this image to create something called a container.

Diagram

Description automatically generated

* A container is an instance of an image, and you can kind of think of it as being like a running program. We're going to go into detail over time, over behind to learn exactly how a container works exactly.
* container is a program with its own isolated set of hardware resources. So it kind of has its own little set or its own little space of memory, has its own little space of networking technology and its own little space of hard drive space as well.
* These images and containers are the absolute backbone of what we are going to working with throughout the rest of this course.



* We can install a piece of software called Docker for Windows, or Docker for Mac depending upon your operating system.
* Inside of this program are two very important tools, The first tool that's inside this package is something called the Docker client.
* The Docker client, also known as the Docker CLI is a program that we are going to interact with quite a bit from our terminal.
* We're going to enter in commands to our terminal, issue them to Docker client.
* Docker client itself doesn't do anything with containers or images. it is just a tool or a portal of sorts to help us interact with another piece of software that is included in this Docker for Windows or Mac package called the Docker server.
* Docker server is frequently called the Docker Damon.
* Docker Damon program is actual tool or the actual piece of software that is responsible for creating containers, images, maintaining containers, uploading images, and doing just about everything you can possibly imagine around the world of Docker.
* So it's the Docker client that you and I issue commands to. It's a thing that we interact with. And behind the scenes, this client is interacting with the Docker server.
* You and I are never going to really reach directly out to this docker server. It's something that's just kind of running it behind the scenes.