

Docker Containers for Open Source Space Software

Artur Scholz
LibreCube Initiative
artur.scholz@librecube.net

What are containers?

Linux containers, in short, contain applications in a way that keep them isolated from the host system that they run on. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package. This makes it easy to move the contained application between environments while retaining full functionality. Think of them as lightweight virtual machines but without the overhead and performance loss due to operating system emulation.

What is "Docker"?

Although containers have been around since decades, their recent popularity has been sparked by the Docker open source project, a command line tool that makes creating and working with containers easy for developers and sysadmins alike.

What is the benefit for you?

Every space mission relies on a ground segment for monitoring and control and some form of data analysis and processing. For open source ground data systems or any other kind of collaborative space software project, the development team is often distributed geographically and in time. Thus it becomes very important that people can reproduce a common development environment without the need of heavy system administration (like setting up databases). Also, the actual deployment of the system may take place in various places and should therefore be as easy and fast as possible.

Docker is ideally suited for DevOps approach, which aims at unifying software development and software operation. DevOps strongly advocates automation and monitoring at all steps of software construction, from integration, testing, releasing to deployment and infrastructure management.

With Docker, the setting up of a development and production is made simple:



Run this for every deployment

```
sudo apt-get update && \
apt-get upgrade -y && \
apt-get install -y python \
git \
python python-dev python-virtualenv \
python-setuptools python-pip \

git clone https://gitlab.com/example.git
cd example
virtualenv venv --python=python3
source venv/bin/activate
pip install -r requirements.txt
...
```



Create Dockerfile once

```
# Use an official Python runtime as a parent image
FROM python:7-slim

# Set the working directory to /app
WORKDIR /app

# Copy the current directory contents into the container at /app
ADD . /app

# Install any needed packages specified in requirements.txt
RUN pip install --trusted-host pypi.python.org -r requirements.txt

# Make port 80 available to the world outside this container
EXPOSE 80

# Define environment variable
ENV NAME World

# Run app.py when the container launches
CMD ["python", "app.py"]
```

Run this for deployment

```
docker build -t friendlyhello .
docker run -p 4000:80 friendlyhello
```

Further info

A number of containers have been prepared in the frame of the LibreCube Initiative. They can be found here:

<https://gitlab.com/libredocker>

Join the mailing list and forum at librecube.net