Saving and Loading Images

An existing image from the Docker cache can be saved to a **tar** file using the **docker save** command. The generated file is not just a regular **tar** file; it contains image metadata and preserves original image layers, so the original image can be later re-created exactly as it was.

The general syntax of the **docker** command **save** verb is:

```
# docker save [-o FILE_NAME] IMAGE_NAME[:TAG]
```

If the **-o** option is not used the generated image is sent to the standard output as binary data.

In the following example, the MySQL container image from Red Hat Software Collections is saved to the file **mysql.tar**:

```
# docker save -o mysql.tar registry.access.redhat.com/rh
scl/mysql-56-rhel7
```

A tar file generated using the **save** verb can be used for backup purposes. To restore the container image, use the **docker load** command. The general syntax of the command is as follows:

```
# docker load [-i FILE_NAME]
```

If the **tar** file given as an argument is *not* a container image with metadata, the **docker load** command will fail.

Following the previous **docker save** example, the image may be restored to the Docker cache using the following command:

```
# docker load -i mysql.tar
```

Note

To save disk space, the file generated by the **save** verb can be gzipped. The **load** verb will automatically **gunzip** the file before importing it to the daemon's cache directory.