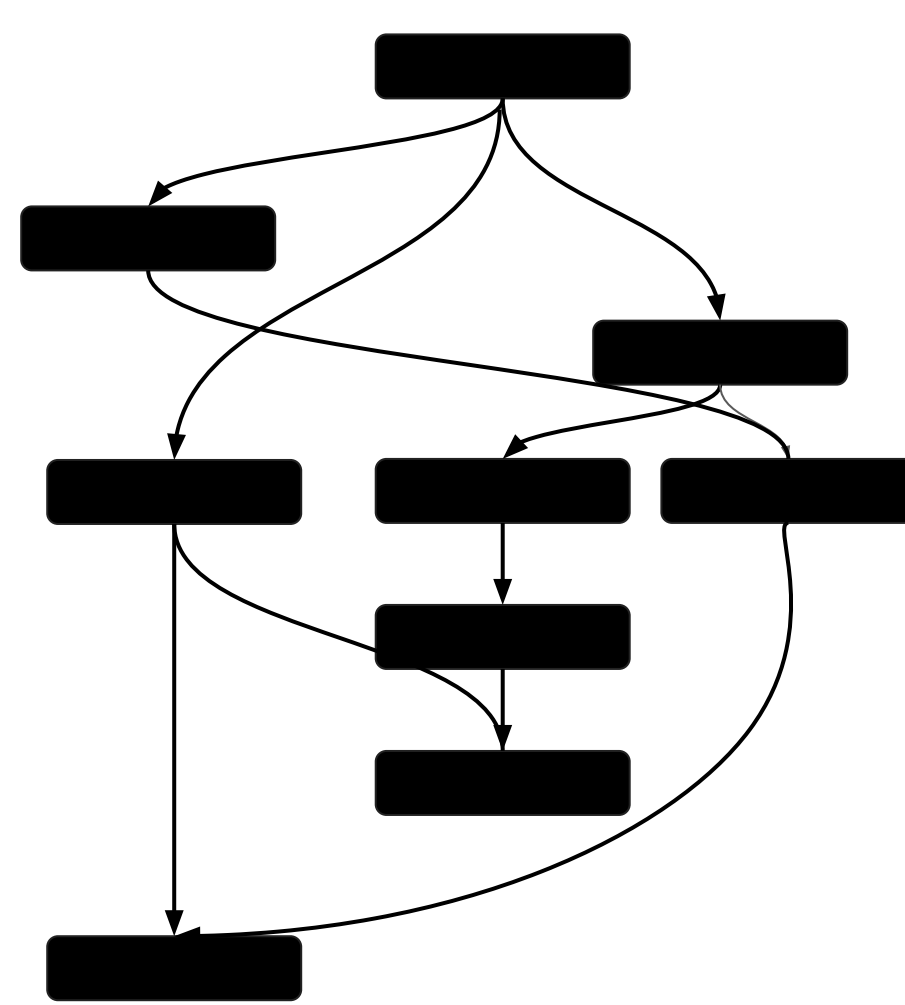


Containers augmented with the Spack package manager can be used to streamline building and deploying scientific applications, providing enhanced portability and competitive performance compared to the bare metal equivalents on both CPUs and GPUs.

Increasing the portability and reproducibility of a scientific application using containers and Spack

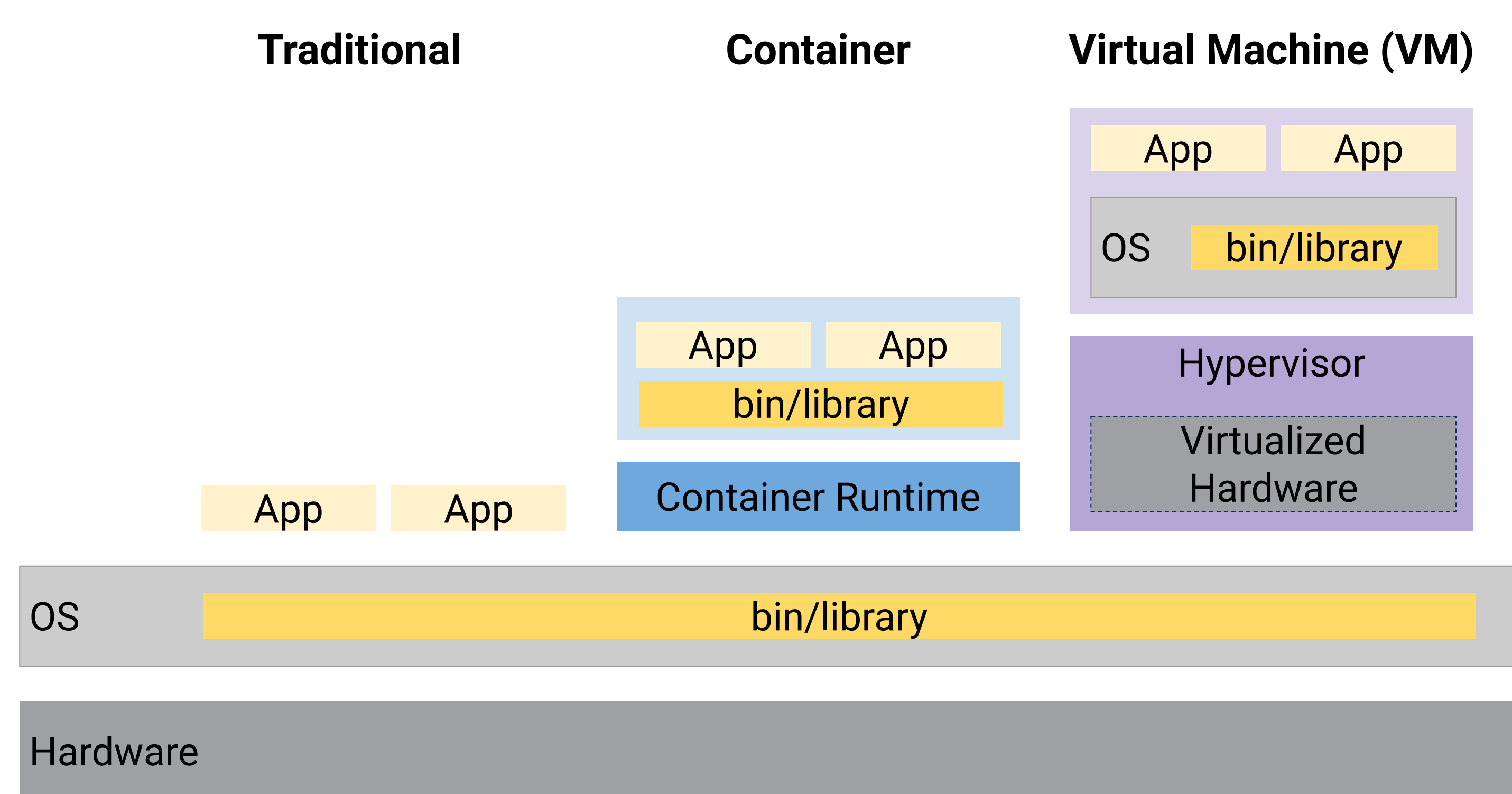
Background

- Manual deployment of scientific apps on new systems can be challenging.
- Spack helps by managing complex package installs while still allowing customization.
- Combining containers and Spack enables both portable and efficient software builds.
- **Samurai**, a scientific application that converts radar data into a wind field, is known to be particularly challenging to build, making it a good candidate for a case-study.



Containers - achieve lightweight portability

Containers bundle applications with a partial operating system.



Spack - install packages the way you want

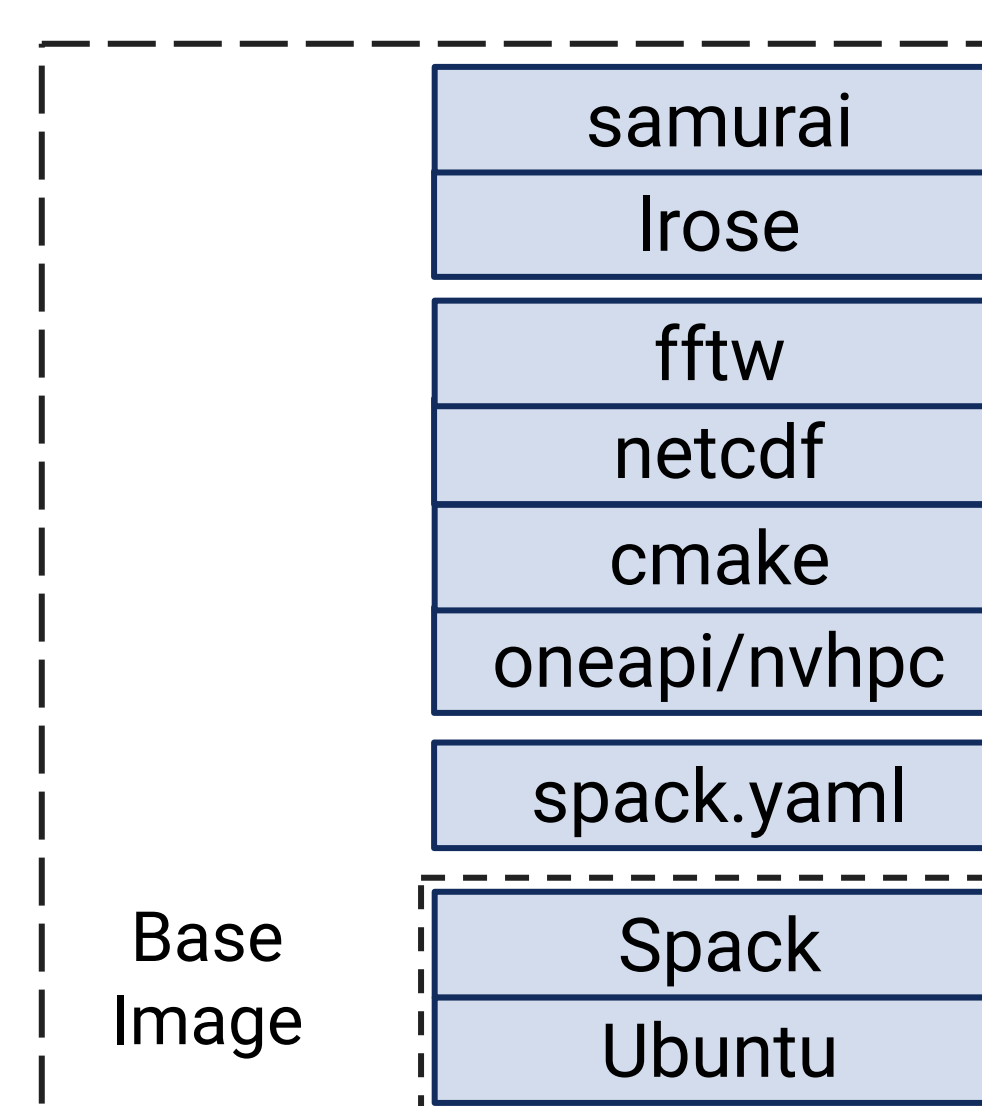
```
$ spack install netcdf-c@4.8.1%nvhpc^hdf5@1.12%gcc
```

Annotations: 'version' points to '4.8.1', 'customized dependencies' points to '^hdf5@1.12%gcc', and 'compiler' points to '%nvhpc'.

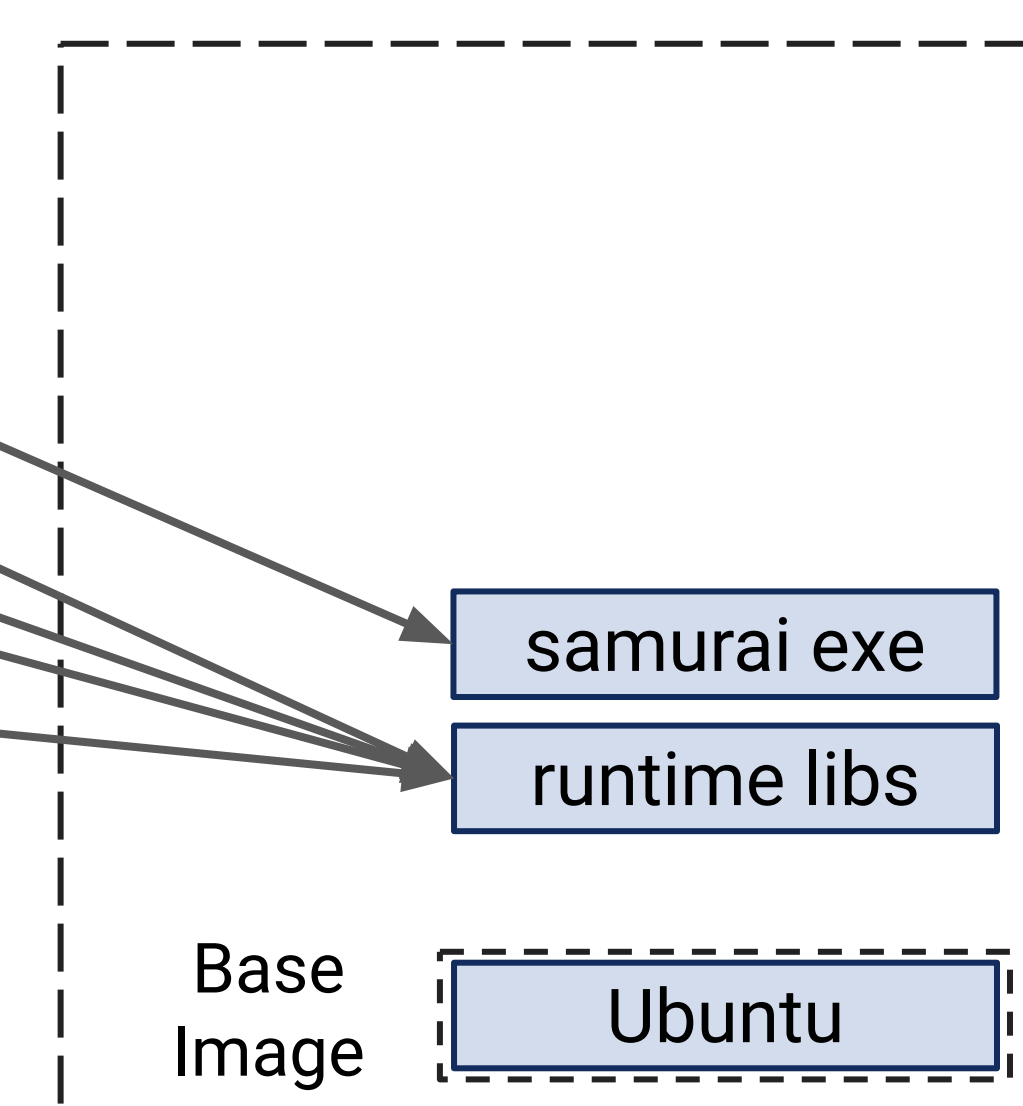
Methods

1. Install compilers and packages in container using Spack
2. Build Irose (dependency) and Samurai in container
3. Make lightweight container
4. Test Samurai on different platforms; compare to bare metal

Developer Container ~10 GB



Lightweight Container ~200 MB



Results: bare metal vs containerized

