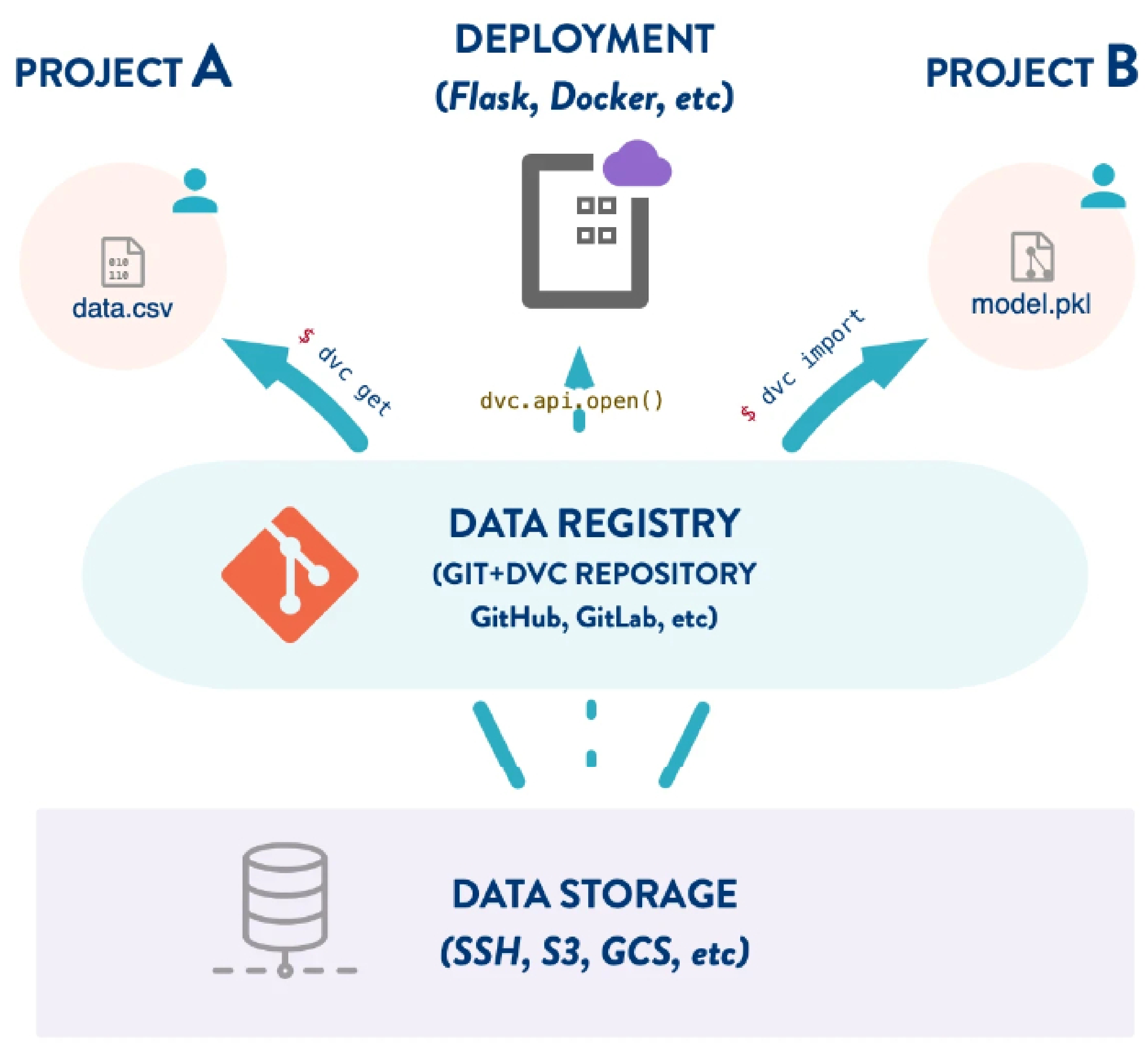
‹ previous
next ›
1 of 2
Automatic Zoom

Create a data registry

Now that we know how to use DVC for data management, we can look at an interesting application: a data registry. This concept allows us to use the same datasets and models across different projects.

In the conceptual model below, we have our data stored in one location. To create a data registry on top of our data storage, we only need a DVC and a Git repository. We can then use DVC's data management features to retrieve the required data from our storage. We can do so from multiple projects at once.



There are numerous benefits to this approach:

- Reusability
- Persistence
- Storage optimization
- Data as code
- Security

We can get data from our data registry in two ways, just like we discussed in lesson 4.7:

```
dvc get <URL> dataset
dvc import <URL> dataset
```

As a refresher, `dvc get` downloads the data from our remote to our directory. `dvc import` does so too and also adds it under DVC control, meaning changes to the data will be tracked.

We could also access the artifacts in our DVC remote from a Python script. This allows us to use models across different projects, for example:

```
import dvc.api.open

model_path = 'model.pkl'
repo_url = 'https://github.com/example/registry'

width dvc.api.open(model_path, repo_url) AS fd:
    model = pickle.load(fd)

# ... Use the model!
```