


Giving Back to the Community

This lesson discusses why we should contribute our custom build packs to the community.

As you saw, creating new build packs is relatively straightforward. In most cases, all we have to do is find the one that is closest to our needs, make a copy, and change a few files. For your internal use, you can configure Jenkins X to use build packs from your own repository. That way you can apply the same workflow to your packs as to any other code you're working on. Moreover, you can import the repository with build packs to Jenkins X and run tests that will validate your changes.

Sometimes you will create packs that are only useful within the context of your company. Most of us think that we have “special” needs, so we tend to have processes and conventions that likely don't fit other organizations. However, more often than not, there is an illusion of being different. The truth is that most of us do employ similar tools, processes, and conventions. The way different companies work and the technologies they use are more alike than many think.

By now, you might be wondering why I'm telling you this. The reason is simple. If you create a build pack, contribute it to the community. You might think that it's specific for your company and only and that it wouldn't be useful to others; that may or may not be true. What is true is that it only takes a few moments to create a pull request. No significant time will be lost if it gets rejected; but if it is merged, many others will benefit from your work, just as you benefit from build packs made by others.

 **This apparently does not matter if the policy of your company does not permit you to make public anything done during your working hours, or if your build pack contains proprietary information.**

Let's wrap up the discussion on custom build packs in the next lesson and remove any used resources.

