

Although I've been comparing IaC tools this entire chapter, the reality is that you will likely need to use multiple tools to build your infrastructure. Each of the tools you've seen has strengths and weaknesses, so it's your job to pick the right tools for the job.

The following sections show three common combinations I've seen work well at a number of companies.

Provisioning plus configuration management

Example: Terraform and Ansible. You use Terraform to deploy all the underlying infrastructure, including the network topology (i.e., virtual private clouds [VPCs], subnets, route tables), data stores (e.g., MySQL, Redis), load balancers, and servers. You then use Ansible to deploy your apps on top of those servers, as depicted in [Figure 1-9](#).

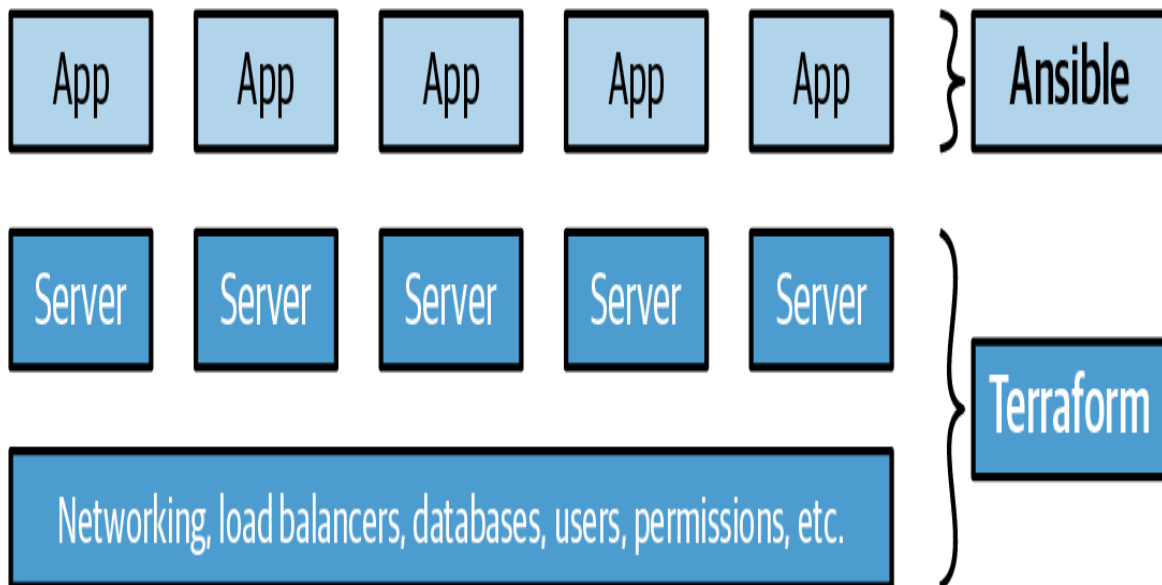


Figure 1-9. Terraform deploys the infrastructure, including servers, and Ansible deploys apps onto those servers.

This is an easy approach to get started with, because there is no extra infrastructure to run (Terraform and Ansible are both client-only applications), and there are many ways to get Ansible and Terraform to work together (e.g., Terraform adds special tags to your servers, and Ansible uses those tags to find the servers and configure them). The major downside is that using Ansible typically means that you're writing a lot of