

The current branch will be updated to reflect the merge, but the target branch will be completely unaffected.

## The Rebase Option

As an alternative to merging, you can rebase the `feature` branch onto `main` branch using the following commands:

```
git checkout feature
git rebase main
```

This moves the entire `feature` branch to begin on the tip of the `main` branch, effectively incorporating all of the new commits in `main`. But, instead of using a merge commit, rebasing *re-writes* the project history by creating brand new commits for each commit in the original branch.

Rebasing the feature branch onto main



`git merge feature`

here is some content not affected by the conflict

<<<<<<< main

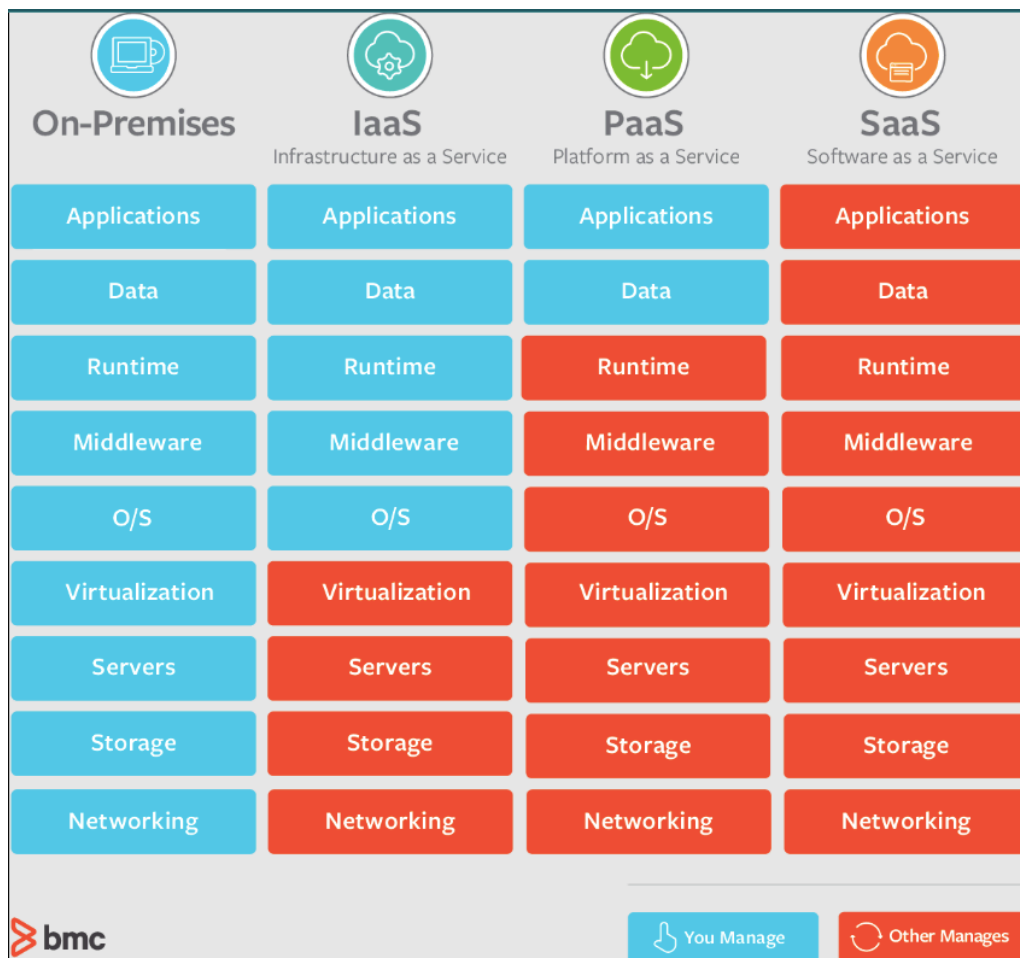
this is conflicted text from main

=====

this is conflicted text from feature branch

>>>>>> feature branch;

Note that merge conflicts will only occur in the event of a 3-way merge.



Platform Type	Common Examples
<b>SaaS</b>	Google Workspace, Dropbox, Salesforce, Cisco WebEx, Concur, GoToMeeting
<b>PaaS</b>	AWS Elastic Beanstalk, Windows Azure, Heroku, Force.com, Google App Engine, Apache Stratos, OpenShift
<b>IaaS</b>	DigitalOcean, Linode, Rackspace, Amazon Web Services (AWS), Cisco Metapod, Microsoft Azure, Google Compute Engine (GCE)