### SVN Inquiry

***Is there an integration with SVN? We are using SVN externals; if we decided to migrate from SVN to Git, how would we go about doing that? What benefits can I expect if my team migrates to Git?***

**Answer**: Thank you so much for reaching out to GitHub. I would be happy to answer your questions.  
  
Yes, there is a way for Git to integrate with SVN. There are different ways you can migrate from SVN to Git and here are those ways:

* 1. *GitHub Importer*: If your existing code is hosted on a public network then you can use the **[GitHub Importer](https://docs.github.com/en/github/importing-your-projects-to-github/about-github-importer)** to import your projects from SVN and here are some easy **[steps](https://docs.github.com/en/github/importing-your-projects-to-github/importing-a-repository-with-github-importer)** to follow for the same. But if your code is hosted on a private network then our tool won’t work. I would recommend you to import using the **[command line](https://docs.github.com/en/github/importing-your-projects-to-github/importing-a-git-repository-using-the-command-line)** for Git repositories.
  2. *git-svn*: When you have migration can be done using **[git-svn](https://github.com/hbt/git-svn-migration)** plus transforming svn externals to Git submodules.
  3. *git-externals*: When it’s impossible to use git submodules then you can use **[git-externals](https://github.com/develersrl/git-externals)** for migration of projects which use svn-externals heavily.

Some of the amazing benefits when you migrate to Git would be as follows:

1. **It’s faster to commit** since you’re mostly working on your local repository and would commit to the central repository every so often. On the other hand when you commit to the central repository more often in SVN, network traffic will slow down everyone
2. **It’s available offline** which means you don’t have to be connected to the central repository all the time like SVN for you to continue working on your project. Your team can keep working offline.
3. **No single point of failure** compared to SVN. In SVN, if your central repository goes down then no developers can commit their changes as well as the code functioning will be disturbed. In Git, every developer has their own repository, so it won’t effect their working if the central repository is broken. They can continue to commit locally and then once the central repository is back up they can push their changes.

### GitHub Enterprise hosted or on-premises

We're considering both the hosted (GitHub Enterprise Cloud) and on-premises (GitHub Enterprise Server) versions of GitHub Enterprise for our Developer team. Which one is the better choice for me to present to management?

**Answer**:

Thank you so much for considering GitHub for your organization.

Both of the options are pretty scalable and very easy to deploy, However, to help you make the right choice, I would like to ask a few questions.

Infrastructure:

1. If you plan to have a on-prem solution, I would recommend take a look at the required specs to see if you have the infrastructure to support this kind of a deployment.
2. Do you have any plans to collaborate with your internal teams in future? Since that will affect which will effect the structure of GitHub Enterprise to collaborate better internally.
3. I would like to make sure if we meet your security and compliance requirements which will have to pass audits as well as something you will be discussing with your management about.

### Technical Git-related question

Hi GitHub! My company has an internal Git server and we also use GitHub.com. For workflow purposes, how do I sync my repository to both GitHub and and the internal Git server?

Note: One common option is git remotes.

Months

January

Feb

March

April

May

June

July