**Installing the development environment**

Follow these instructions to obtain the Hyperledger Composer development tools (primarily used to *create* Business Networks) and stand up a Hyperledger Fabric (primarily used to *run/deploy* your Business Networks locally). Note that the Business Networks you create can also be deployed to Hyperledger Fabric runtimes in other environments e.g. on a cloud platform.

**Installing components**

**Step 0: Install Docker**

Please follow these steps: <https://docs.docker.com/toolbox/toolbox_install_windows/#step-2-install-docker-toolbox>

**Step 1: Install the CLI tools**

There are a few useful CLI tools for Composer developers. The most important one is composer-cli, which contains all the essential operations, so we'll install that first. Next, we'll also pick up generator-hyperledger-composer, composer-rest-server and Yeoman plus the generator-hyperledger-composer. Those last 3 are not core parts of the development environment, but they'll be useful if you're following the tutorials or developing applications that interact with your Business Network, so we'll get them installed now.

Note that you **should not** use su or sudo for the following npm commands.

1. Essential CLI tools:

Copy

npm install -g composer-cli

1. Utility for running a REST Server on your machine to expose your business networks as RESTful APIs:

Copy

npm install -g composer-rest-server

1. Useful utility for generating application assets:

Copy

npm install -g generator-hyperledger-composer

1. Yeoman is a tool for generating applications, which utilises generator-hyperledger-composer:

Copy

npm install -g yo

**Step 2: Install Playground**

If you've already tried Composer online, you'll have seen the browser app "Playground". You can run this locally on your development machine too, giving you a UI for viewing and demonstrating your business networks.

1. Browser app for simple editing and testing Business Networks:

Copy

npm install -g composer-playground

**Step 3: Set up your IDE**

Whilst the browser app *can* be used to work on your Business Network code, most users will prefer to work in an IDE. Our favourite is VSCode, because a Composer extension is available.

1. Install VSCode from this URL: <https://code.visualstudio.com/download>
2. Open VSCode, go to Extensions, then search for and install the Hyperledger Composer extension from the Marketplace.

**Step 4: Install Hyperledger Fabric**

This step gives you a local Hyperledger Fabric runtime to deploy your business networks to. Please run these steps as an ADMIN

1. In a directory of your choice (we will assume ~/fabric-dev-servers), get the .tar.gz file that contains the tools to install Hyperledger Fabric :

Copy

md fabric-dev-servers

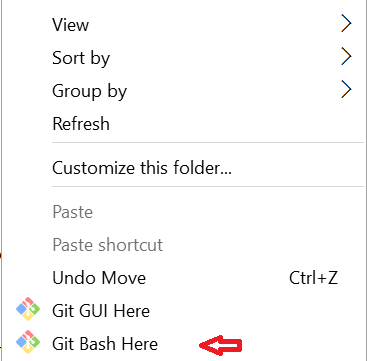
cd fabric-dev-servers

choco install curl

curl -O https://raw.githubusercontent.com/hyperledger/composer-tools/master/packages/fabric-dev-servers/fabric-dev-servers.zip

unzip fabric-dev-servers.zip

1. Install [GIT](https://git-scm.com/downloads). During installation of GIT, add GIT Bash to windows context menu by selecting its option. After installation right click in your folder select GIT Bash Here and use your sh command like for example:



1. Use the scripts you just downloaded and extracted to download a local Hyperledger Fabric runtime:

Copy

cd ~/fabric-dev-servers

./downloadFabric.sh

Congratulations, you've now installed everything required for the typical Developer Environment. Read on to learn some of the most common things you'll do with this environment to develop and test your Blockchain Business Networks.

**Controlling your dev environment**

**Starting and stopping Hyperledger Fabric**

You control your runtime using a set of scripts which you'll find in ~/fabric-dev-servers if you followed the suggested defaults.

The first time you start up a new runtime, you'll need to run the start script, then generate a PeerAdmin card:

Copy

cd ~/fabric-dev-servers

./startFabric.sh

./createPeerAdminCard.sh

You can start and stop your runtime using ~/fabric-dev-servers/stopFabric.sh, and start it again with ~/fabric-dev-servers/startFabric.sh.

At the end of your development session, you run ~/fabric-dev-servers/stopFabric.sh and then ~/fabric-dev-servers/teardownFabric.sh. Note that if you've run the teardown script, the next time you start the runtime, you'll need to create a new PeerAdmin card just like you did on first time startup.

The local runtime is intended to be frequently started, stopped and torn down, for development use. If you're looking for a runtime with more persistent state, you'll want to run one outside of the dev environment, and deploy Business Networks to it. Examples of this include running it via Kubernetes, or on a managed platform such as IBM Cloud.

**Start the web app ("Playground")**

To start the web app, run:

Copy

composer-playground

It will typically open your browser automatically, at the following address: <http://localhost:8080/login>

You should see the PeerAdmin@hlfv1 Card you created with the createPeerAdminCard script on your "My Business Networks" screen in the web app: if you don't see this, you may not have correctly started up your runtime!

Congratulations, you've got all the components running, and you also know how to stop and tear them down when you're done with your dev session.