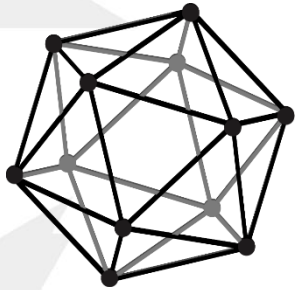


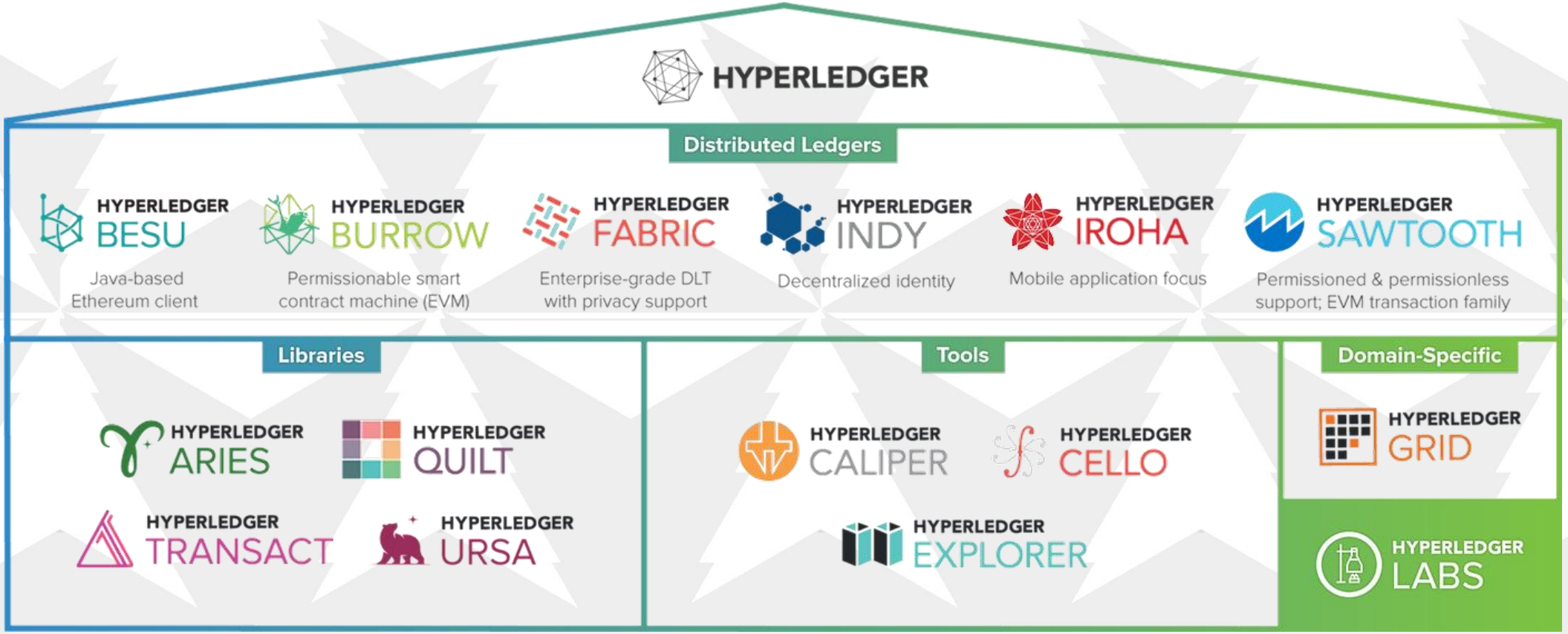
Hyperledger

Dr. Sarwan Singh

NIELIT Chandigarh

Agenda

- Hyperledger – introduction, History
-



References

- hyperledger.org - Whitepaper_IntroductiontoHyperledger
- medium.com/hackernoon

Introduction

- Hyperledger is an open source collaborative effort to advance cross-industry open standard for distributed ledgers that can transform the way business transactions are conducted globally
- Global collaboration spanning finance, banking, IoT, supply chains, healthcare, manufacturing, technology and more.
- The Linux Foundation hosts Hyperledger under the foundation
- Hyperledger operates under an Apache 2.0 license for code
- Creative Commons Attribution 4.0 International license for content

Source : hyperledger.org

History Timeline

Introduced on 9th Feb 2016 in San Francisco, California by Linux foundation

30 founding members including VMware, ConsenSys, IBM, etc.

They set an Open Technical Governance Structure

Right now hyperledger has 250+ members

Digital Asset donates the name Hyperledger to the Linux Foundation

Executive director of Hyperledger — **Brian Behlendorf**

Behlendorf was a primary developer of the Apache Web server and a founding member of the Apache Software Foundation

Apache Web server – Primary developer

Apache Software Foundation – Founding member

Mozilla Foundation – BOD since 2003

Electronic Frontier Foundation – BOD since 2013

CollabNet – Founding CTO

World Economic Forum – CTO

Most recently, managing director at **Mithril Capital Management LLC**, a global technology investment firm

Open source is popular and reliable

- Linux operating system runs 90% of the public cloud workload,
- more than 80% of the world's smartphones, and
- 99% of all supercomputers.
- The open source Apache web server has been the world's most popular web server for more than 20 years, and today supports more than 40% of all active websites
- mySQL—the world's most popular database server
- the Firefox web browser.

THE LINUX FOUNDATION

Together with the global technology community, The Linux Foundation® is solving the world's hardest problems through open source and **creating the largest shared technology investment in history**.

With 16 years experience providing **governance structure, IT infrastructure and ecosystem development**, The Linux Foundation is the umbrella organization for **more than 60 open source projects** accelerating open technology development and commercial adoption.

Some of the game-changing initiatives hosted by The Linux Foundation include:

OPENCHAIN, Let's Encrypt, JS Foundation, OPEN CONTAINER INITIATIVE, CLOUD NATIVE COMPUTING FOUNDATION, Xen Project, Core Embedded Linux Project, OPEN DAYLIGHT, CORD, CORE INFRASTRUCTURE INITIATIVE, OPNFV, CLOUD FOUNDRY, node, yocto PROJECT

Hyperledger Modular Umbrella Approach

Infrastructure

Technical, Legal, Marketing, Organizational

Ecosystems that accelerate open development and commercial adoption

THE LINUX FOUNDATION

Cloud Foundry | Node.js | **Hyperledger** | Open Container Initiative

Frameworks

Meaningfully differentiated approaches to business blockchain frameworks developed by a growing community of communities

Hyperledger **Indy** | Hyperledger **Fabric** | Hyperledger **Iroha** | Hyperledger **Sawtooth** | Hyperledger **Burrow**

Tools

Typically built for one framework, and through common license and community of communities approach, ported to other frameworks

Hyperledger **Quilt** | Hyperledger **Composer** | Hyperledger **Explorer** | Hyperledger **Cello**

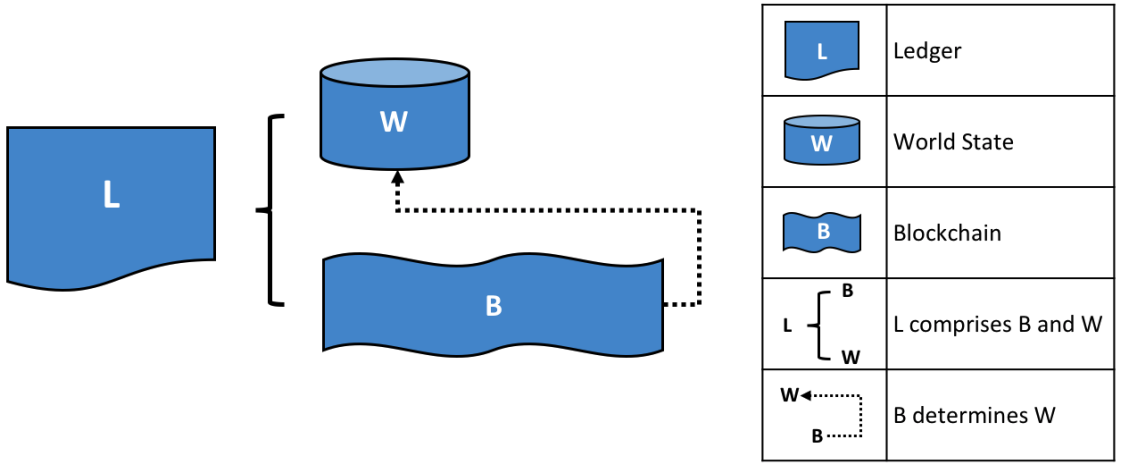
Hyperledger Business Blockchain Frameworks

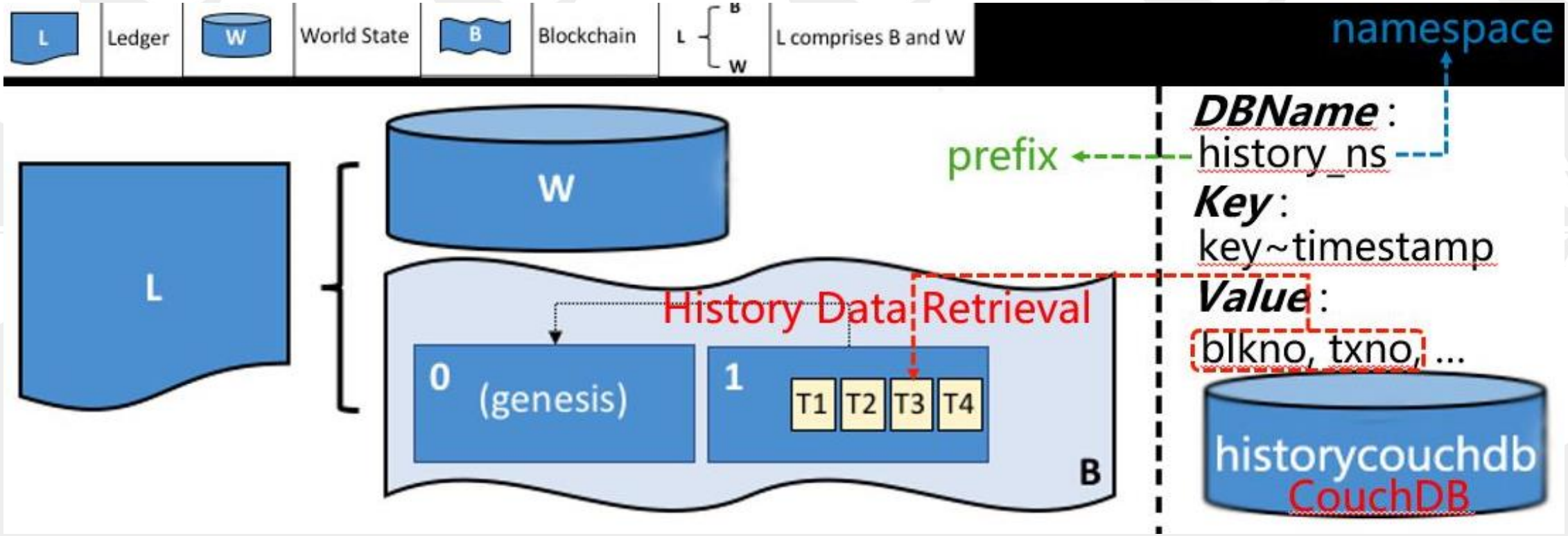
- **Hyperledger Fabric**: Intended as a foundation for developing applications or solutions with a modular architecture, Hyperledger Fabric allows components, such as consensus and membership services, to be plug-and-play.
- **Hyperledger Iroha**: A business blockchain framework designed to be simple and easy to incorporate into infrastructural projects requiring distributed ledger technology.
- **Hyperledger Sawtooth**: A modular platform for building, deploying, and running distributed ledgers. Hyperledger Sawtooth includes a novel consensus algorithm, Proof of Elapsed Time (PoET), which targets large distributed validator populations with minimal resource consumption.
- **Hyperledger Burrow**: A permissionable smart contract machine. The first of its kind when released in December, 2014, Burrow provides a modular blockchain client with a permissioned smart contract interpreter built in part to the specification of the Ethereum Virtual Machine (EVM).
- **Hyperledger Indy**: Tools, libraries, and reusable components for providing digital identities rooted on blockchains or other distributed ledgers so that they are interoperable across administrative domains, applications, and any other silo.

Hyperledger Design Philosophy

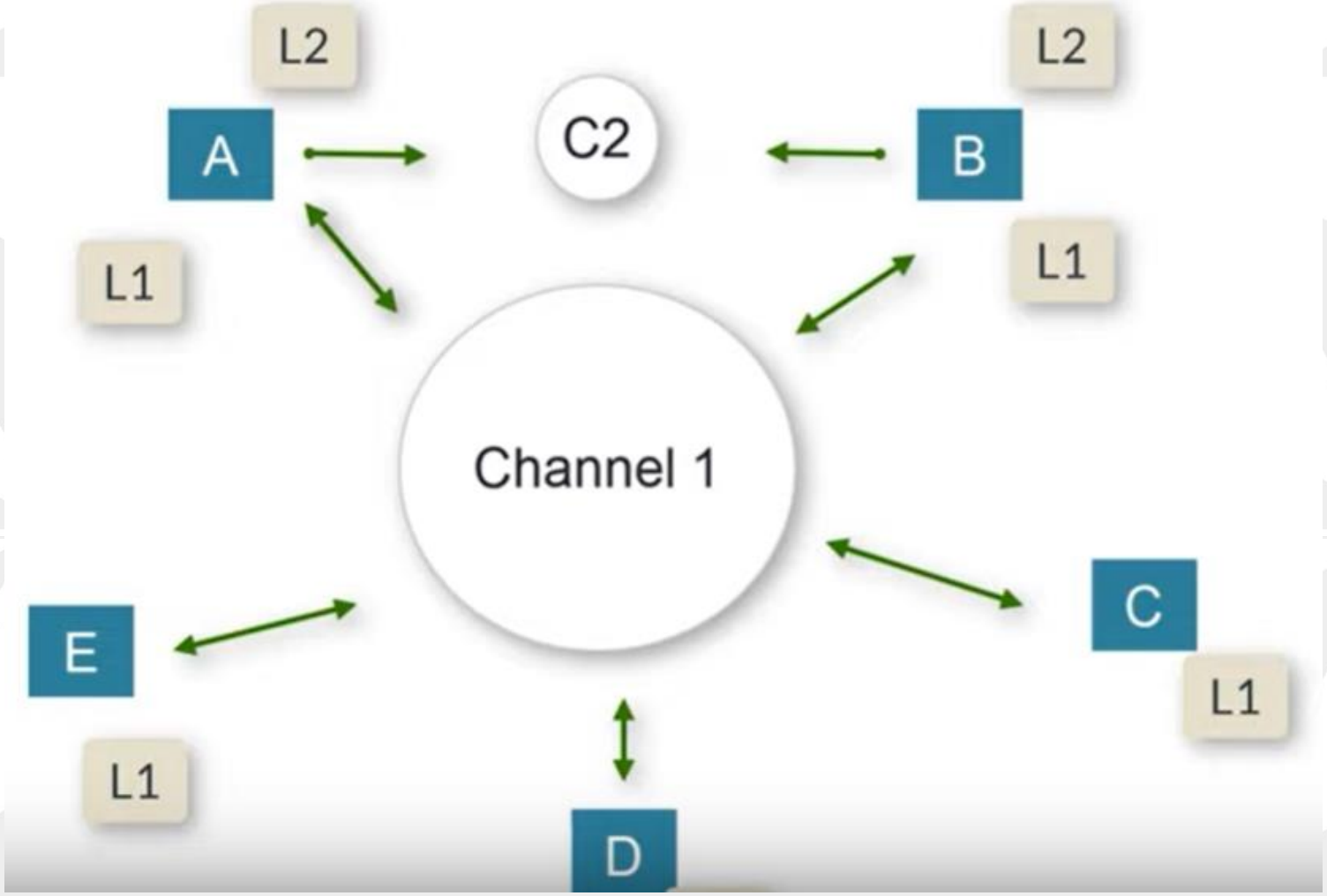
About Hyperledger

- Hyperledger is essentially broken into three categories,
  - the Distributed Ledger Technology (DLT), basically blockchain,
  - the libraries, and
  - the tools.
- Channels
- World state
- Ledger



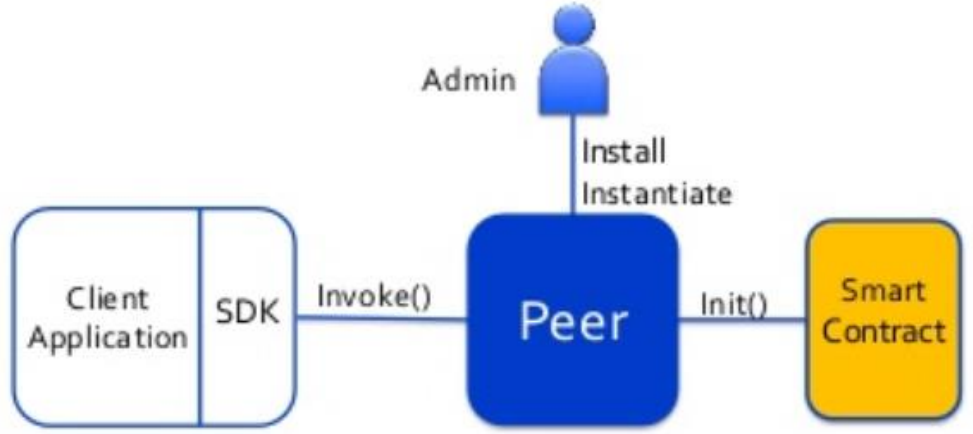

L
Ledger
W
World State
B
Blockchain
L
B
W
L comprises B and W
namespace
DBName :
history_ns
prefix
Key :
key~timestamp
Value :
blkno, txno, ...
W
L
History Data Retrieval
0
(genesis)
1
T1
T2
T3
T4
B
historycouchdb
CouchDB

Hyperledger – Channel



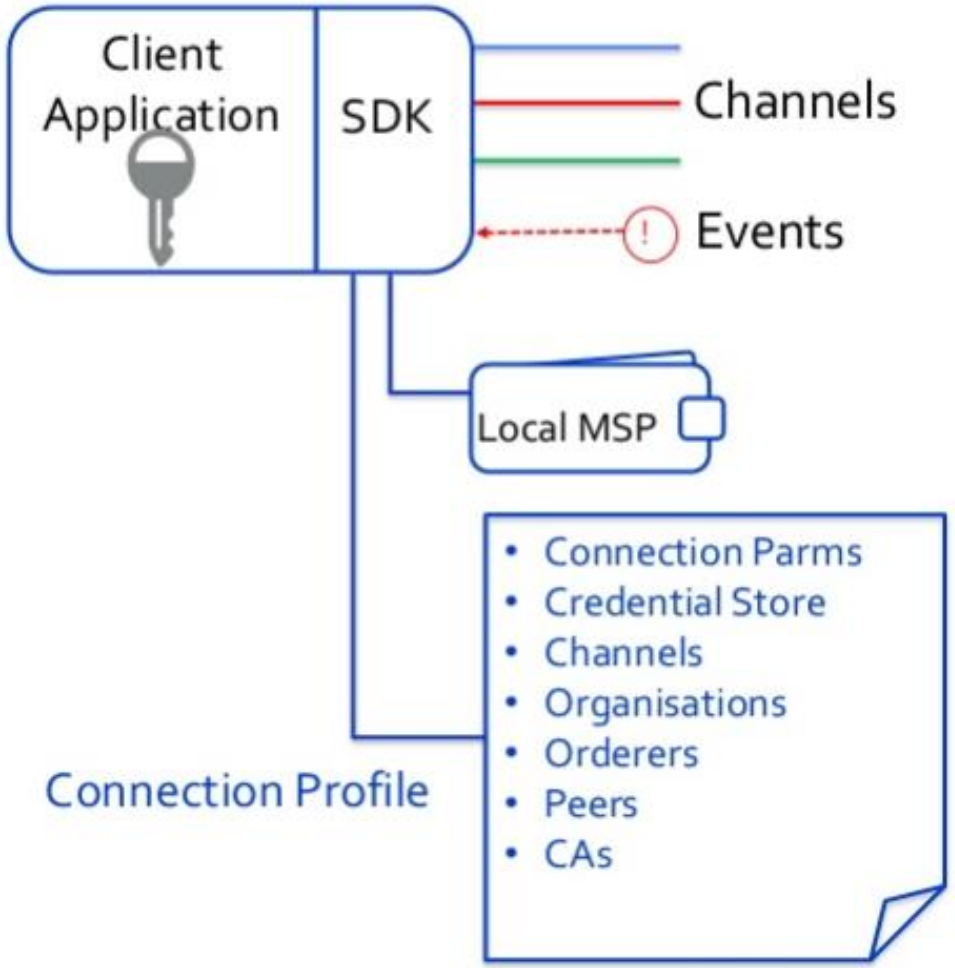
Smart Contact

- Smart contact contain business logic deployed to peers
- Interact with the world state through the Fabric interface
- Language support for :
  - Golang
  - Node.js
  - Java

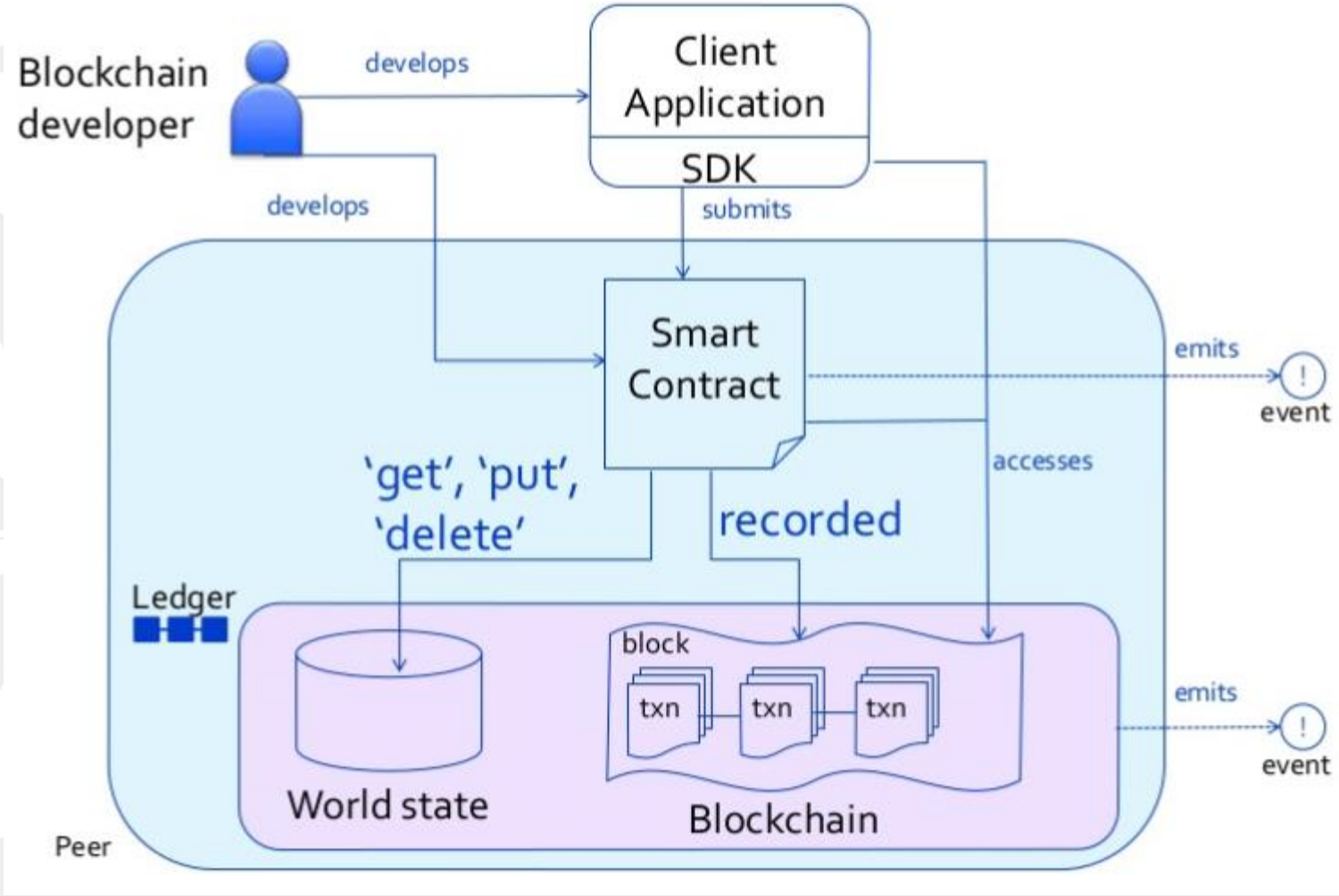


Client Application

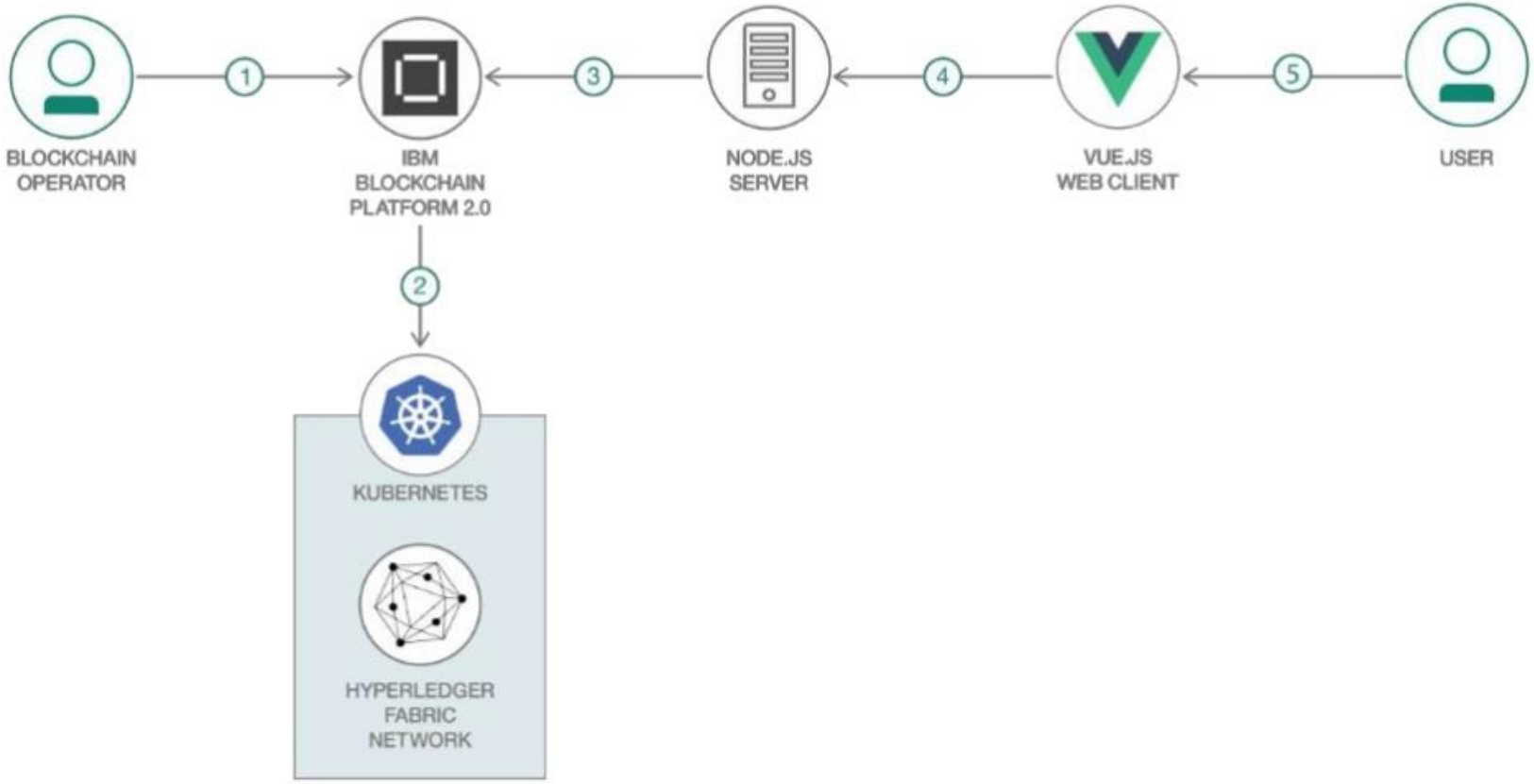
- Client application use Fabric SDK to :
  - Connect over channels to peer and order nodes
  - Provide public / private keys
- Connection Profile
  - Network end points and connection parameters
  - The gateway to submit transactions to a Hyperledger Fabric Network



How application interact with Ledger



App Architecture

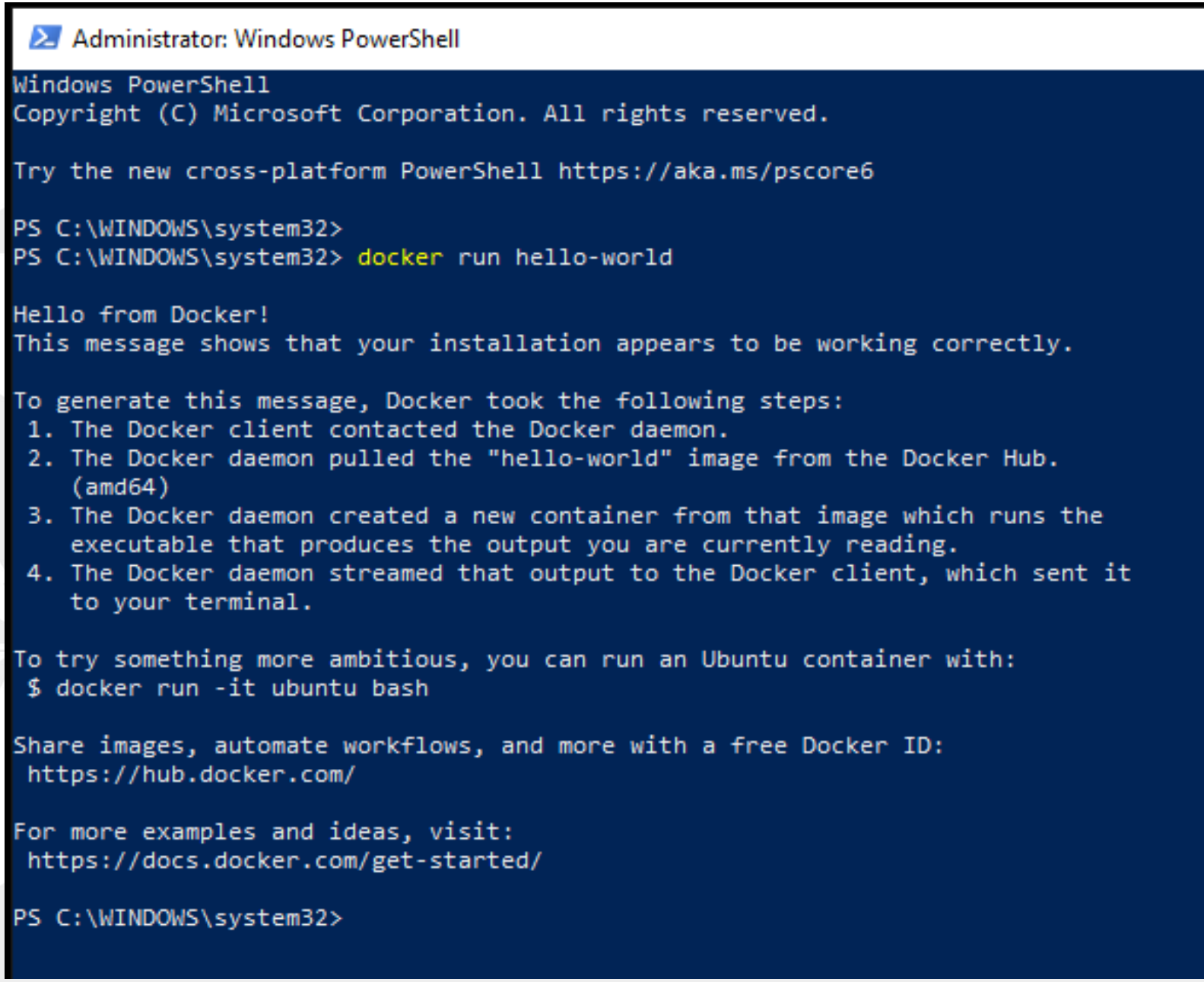


Installation Steps

- CURL
- Docker
- Docker-Compose
- Golang
- Node.js and NPM
- Python 2.7

Docker

- Hello world program



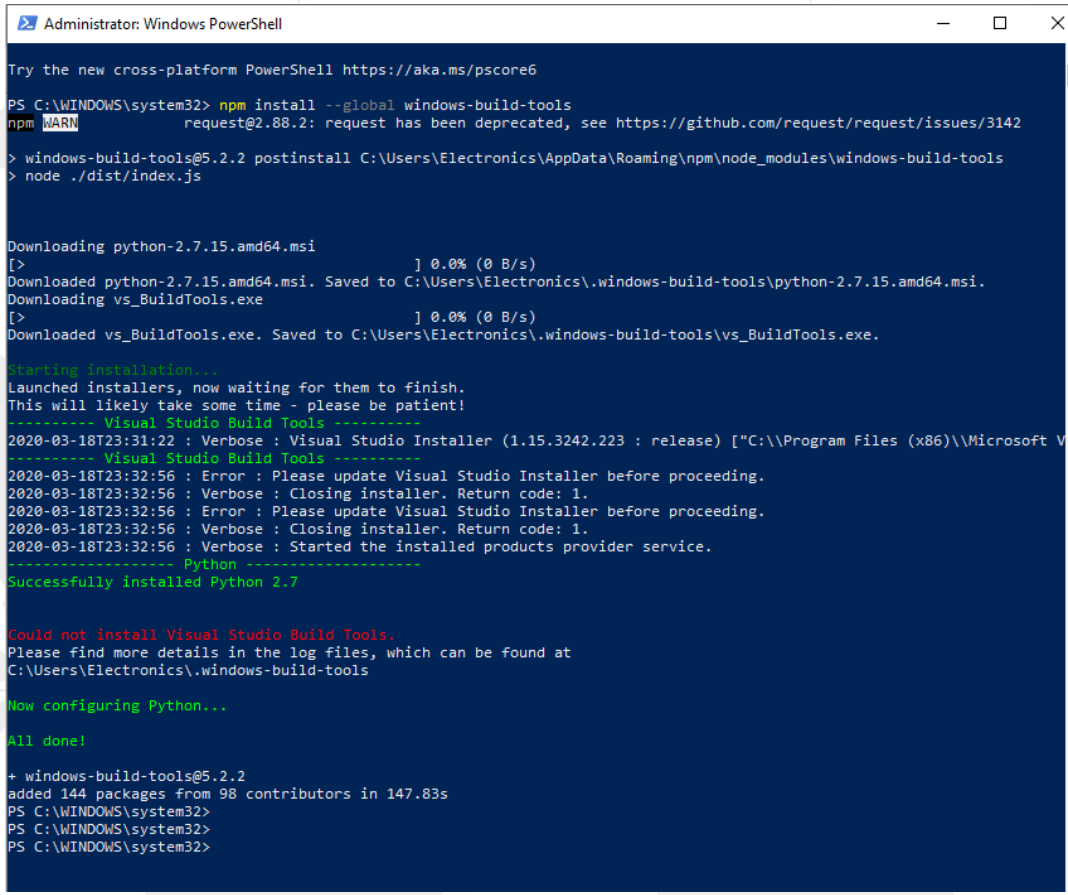
Installation Steps

Install windows-build-tools and grpc

- npm install **--global** windows-build-tools
- npm install **--global** grpc

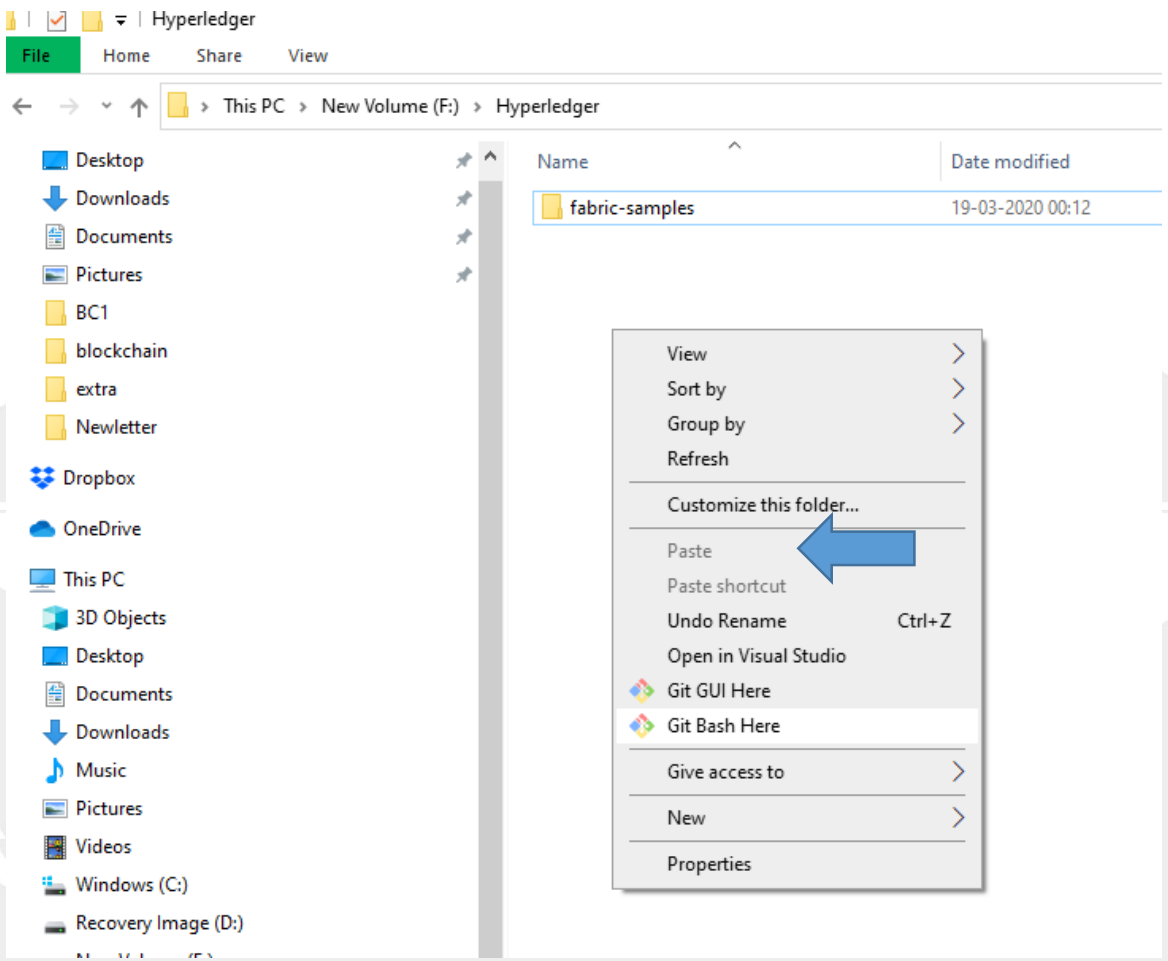
Install git to run the bash commands

- Git config –global core.autocrlf false
- Git config –global core.longpaths true



Install Samples, Binaries and Docker Images

- Run Script $ curl -sSL http://bit.ly/2ysbOFE | bash -s




```
Electronics@DESKTOP-JMQB6VJ MINGW64 /f/Hyperledger
$

Electronics@DESKTOP-JMQB6VJ MINGW64 /f/Hyperledger
$ curl -sSL http://bit.ly/2ysbOFE | bash -s

Clone hyperledger/fabric-samples repo

===> Cloning hyperledger/fabric-samples repo and checkout v2.0.0
Cloning into 'fabric-samples'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4820 (delta 0), reused 0 (delta 0), pack-reused 4816
Receiving objects: 100% (4820/4820), 1.71 MiB | 553.00 KiB/s, done.
Resolving deltas: 100% (2428/2428), done.
error: pathspec 'v2.0.0' did not match any file(s) known to git

Pull Hyperledger Fabric binaries

===> Downloading version 2.0.0 platform specific fabric binaries
===> Downloading:  https://github.com/hyperledger/fabric/releases/download/v2.0.
0/hyperledger-fabric-windows-amd64-2.0.0.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   635  100   635    0     0    755      0 --:--:-- --:--:-- --:--:--   755
100 64.3M  100 64.3M    0     0  27687      0  0:40:36  0:40:36 --:--:-- 34747
==> Done.
===> Downloading version 1.4.6 platform specific fabric-ca-client binary
===> Downloading:  https://github.com/hyperledger/fabric-ca/releases/download/v1
.4.6/hyperledger-fabric-ca-windows-amd64-1.4.6.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   638  100   638    0     0    728      0 --:--:-- --:--:-- --:--:--   728
 90 19.1M   90 17.3M    0     0  19757      0  0:16:56  0:15:18  0:01:38 23072
```

Test the fabric netwo

- cd fabric-samples/first-networ

- ./byfn.sh up
- ./byfn.sh generate
- ./byfn.sh down

```
Creating network "net_byfn" with the default driver
Creating volume "net_orderer.example.com" with default driver
Creating volume "net_peer0.org1.example.com" with default driver
Creating volume "net_peer1.org1.example.com" with default driver
Creating volume "net_peer0.org2.example.com" with default driver
Creating volume "net_peer1.org2.example.com" with default driver
Pulling orderer.example.com (hyperledger/fabric-orderer:latest)...
latest: Pulling from hyperledger/fabric-orderer
Pulling peer0.org1.example.com (hyperledger/fabric-peer:latest)...
latest: Pulling from hyperledger/fabric-peer
Creating peer1.org2.example.com ... done
Creating peer0.org2.example.com ... done
Creating peer1.org1.example.com ... done
Creating orderer.example.com    ... done
Creating peer0.org1.example.com ... done
Creating cli                    ... done

 ____    _____      _      ____    _____
/ ___|  |_   _|    / \    |  _ \  |_   _|
\___ \    | |     / _ \   | |_) |   | |
 ___) |   | |    / ___ \  |  _ <    | |
|____/    |_|   /_/   \_\ |_| \_\   |_|

Build your first network (BYFN) end-to-end test

Channel name : mychannel
Creating channel...
```

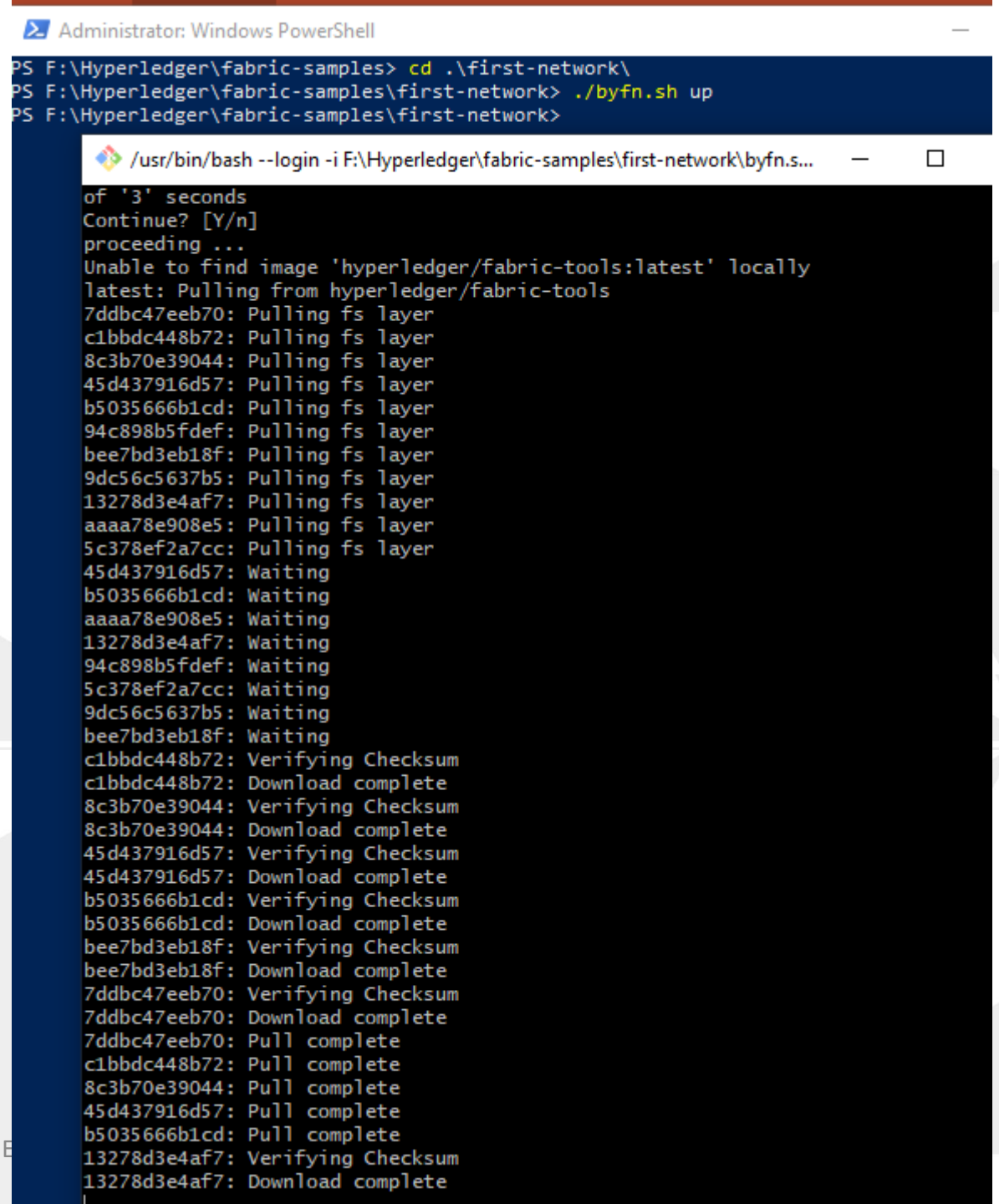
```
Querying chaincode on peer1.org2...
===================== Querying on peer1.org2 on channel 'mychannel'... ===
+ peer chaincode query -C mychannel -n mycc -c '{"Args":["query","a"]}'
Attempting to Query peer1.org2 ...3 secs
+ res=0
+ set +x

90
===================== Query successful on peer1.org2 on channel 'mychannel' ==

========= All GOOD, BYFN execution completed ===========


 _____   _   _   ____
| ____| | \ | | |  _ \
|  _|   |  \| | | | | |
| |___  | |\  | | |_| |
|_____| |_| \_| |____/
```

- ./byfn.sh up


Administrator: Windows PowerShell
PS F:\Hyperledger\fabric-samples> cd .\first-network\
PS F:\Hyperledger\fabric-samples\first-network> ./byfn.sh up
PS F:\Hyperledger\fabric-samples\first-network>
/usr/bin/bash --login -i F:\Hyperledger\fabric-samples\first-network\byfn.s...
of '3' seconds
Continue? [Y/n]
proceeding ...
Unable to find image 'hyperledger/fabric-tools:latest' locally
latest: Pulling from hyperledger/fabric-tools
7ddbc47eeb70: Pulling fs layer
c1bbdc448b72: Pulling fs layer
8c3b70e39044: Pulling fs layer
45d437916d57: Pulling fs layer
b5035666b1cd: Pulling fs layer
94c898b5fdef: Pulling fs layer
bee7bd3eb18f: Pulling fs layer
9dc56c5637b5: Pulling fs layer
13278d3e4af7: Pulling fs layer
aaaa78e908e5: Pulling fs layer
5c378ef2a7cc: Pulling fs layer
45d437916d57: Waiting
b5035666b1cd: Waiting
aaaa78e908e5: Waiting
13278d3e4af7: Waiting
94c898b5fdef: Waiting
5c378ef2a7cc: Waiting
9dc56c5637b5: Waiting
bee7bd3eb18f: Waiting
c1bbdc448b72: Verifying Checksum
c1bbdc448b72: Download complete
8c3b70e39044: Verifying Checksum
8c3b70e39044: Download complete
45d437916d57: Verifying Checksum
45d437916d57: Download complete
b5035666b1cd: Verifying Checksum
b5035666b1cd: Download complete
bee7bd3eb18f: Verifying Checksum
bee7bd3eb18f: Download complete
7ddbc47eeb70: Verifying Checksum
7ddbc47eeb70: Download complete
7ddbc47eeb70: Pull complete
c1bbdc448b72: Pull complete
8c3b70e39044: Pull complete
45d437916d57: Pull complete
b5035666b1cd: Pull complete
13278d3e4af7: Verifying Checksum
13278d3e4af7: Download complete